Л.В. Чепак, А.Г. Масловская

Численные методы. Использование Matlab

Часть I

Учебно-методическое пособие

Министерство образования и науки Российской Федерации
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Факультет математики и информатики

Л.В. Чепак, А.Г. Масловская

Численные методы. Использование Matlab
Ч. I

Учебно-методическое пособие для студ. спец. 010400, 010200
010100, 220200 , 071900

Благовещенск
2005

ББК
Т80

Печатается по решению редакционно-издательского совета факультета математики и информатики Амурского государственного университета

Чепак Л.В., Масловская А.Г.

Численные методы. Использование Matlab Ч.I. Учебно-методическое пособие . – Благовещенск: Изд-во АмГУ,2005

Аннотация

В данном методическом пособии, ориентированном на пакет Matlab, изложены методы численного анализа: численные решения нелинейных уравнений и систем, систем линейных уравнений, теория интерполирования, численное дифференцирование и интегрирование, использование численных методов для обработки экспериментальных данных. Все методы иллюстрируются примерами, в которых используются программы, реализованные в пакете Matlab. Приводятся также варианты индивидуальных заданий к лабораторному практикуму.

Пособие предназначено для студентов специальностей 010100 – «математика», 010200 – «прикладная математика», 010400 – «физика», 220200 – «автоматизированные системы обработки информации и управления», 071900 – «информационные системы в технике и технологиях».

Рецензенты:

Содержание

Введение

Численные методы занимают важное место в системе прикладного математического образования. Данный курс тесно связан с основными математическими дисциплинами специальностей 010100 – «математика», 010200 – «прикладная математика», 010400 – «физика», 220200 – «автоматизированные системы обработки информации и управления», 071900 – «информационные системы в технике и технологиях»: линейная алгебра, математический анализ, дифференциальные уравнения, уравнения в частных производных, алгоритмические языки, практикум на ЭВМ.

Данное учебно-методическое пособие содержит материал для организации практического и лабораторного практикума по следующим разделам, предусмотренным государственным образовательным стандартом: численные решения нелинейных уравнений и их систем, систем линейных уравнений, теория интерполирования, численное дифференцирование и интегрирование, применение численных методов к обработке экспериментальных данных. Здесь приводятся необходимые теоретические сведения, подробные алгоритмы типовых методов, их программные реализации и соответствующие функции математического пакета Matlab, контрольные вопросы для самопроверки, а также индивидуальные задания к лабораторному практикуму.

Основной особенностью пособия является его прикладная направленность. Студенты должны не только правильно выбрать подход к решению конкретной задачи, сформулировать математическое содержание конкретного метода (границ его применимости, погрешности метода и т.д.), а также проявить умение использовать современные программные средства.

§ 1 Решение нелинейных уравнений

Пусть задана функция $f(x)$. Необходимо найти корни уравнения

$$f(x) = 0. \qquad (1)$$

Вычислить с заданной точностью ε корень уравнения $\bar{x}$, принадлежащий отрезку $[a, b]$. Предполагается, что на отрезке $[a, b]$ находится единственный корень, причем функция $f(x)$ на данном отрезке непрерывна.

1.1 Метод половинного деления

Разделим отрезок $[a, b]$ пополам точкой $c = \frac{a+b}{2}$. Если $f(c) \neq 0$, то возможны два случая:

– функция $f(x)$ меняет знак на отрезке $[a, c]$;

– функция $f(x)$ меняет знак на отрезке $[c, b]$.

Выбирая в каждом случае тот отрезок, на котором функция меняет знак, и продолжаем процесс половинного деления до тех пор, пока не выполнится условие

$$\frac{b-a}{2^n} \leq \varepsilon,$$

где n – число проведенных делений исходного отрезка пополам. После этого любую из границ последнего отрезка можно взять в качестве корня уравнения.

Рисунок 1 иллюстрирует алгоритм нахождение корня методом половинного деления.

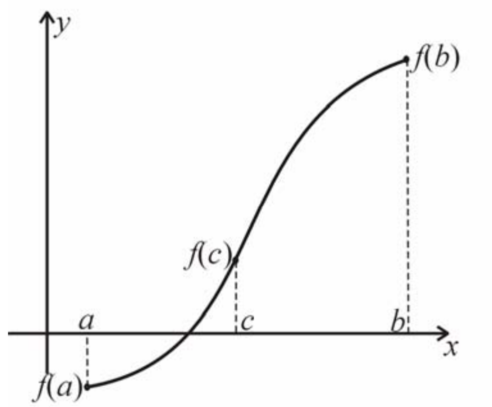


Рисунок 1 – Приближение к корню уравнения методом половинного деления

1.2 Метод секущих

Фиксируется некоторая точка $x_0 \in [a,b]$, в которой $f(x_0) \cdot f''(x_0) > 0$. Начальным приближением корня уравнения (1) выбирается некоторая точка $x_1 \in [a,b]$, удовлетворяющая условию $f(x_0) \cdot f(x_1) < 0$. Последующие приближения находятся следующим образом:

$$x_n = \frac{x_0 \cdot f(x_{n-1}) - x_{n-1} \cdot f(x_0)}{f(x_{n-1}) - f(x_0)}, \quad n = 2,3,... \qquad (2)$$

Достижение заданной точности ε на каждом шаге можно определить по формуле

$$\left|\bar{x} - x_n\right| \le \frac{\left|f(x_n)\right|}{\alpha},$$

где $\alpha = \min\limits_{x \in [x_0, x_1]} \left|f'(x)\right|$.

Рисунок 2 иллюстрирует алгоритм нахождение корня методом секущих.

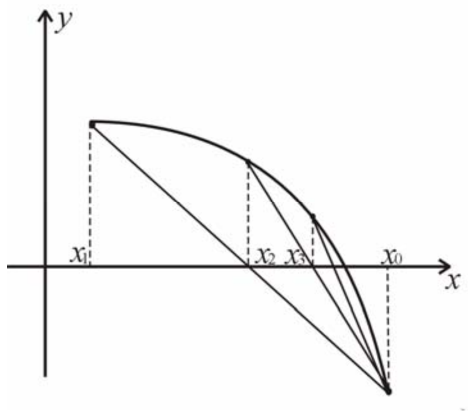


Рисунок 2 – Приближение к корню уравнения методом секущих

1.3 Метод касательных

Функция $f(x)$ имеет непрерывные производные $f'(x)$, $f''(x)$, не обращающиеся в нуль на отрезке $[a,b]$. Зададим некоторое начальное приближение вычисляемого корня $x_0 \in [a,b]$ так, чтобы $f(x_0) \cdot f''(x_0) > 0$. Вычисляем последовательности приближений к искомому решению

$$x_n = x_{n-1} - \frac{f(x_{n-1})}{f'(x_{n-1})}, \quad n = 1,2,... \qquad (3)$$

Скорость сходимости метода касательных можно оценить следующим образом:

$$\left|x_n - \bar{x}\right| \le \frac{\beta}{2 \cdot \gamma}\left|x_{n-1} - \bar{x}\right|^2,$$

где $\beta = \max\limits_{[a,b]}\left|f''(x)\right|$, $\gamma = \min\limits_{[a,b]}\left|f'(x)\right|$.

Рисунок 3 иллюстрирует алгоритм нахождение корня методом касательных.

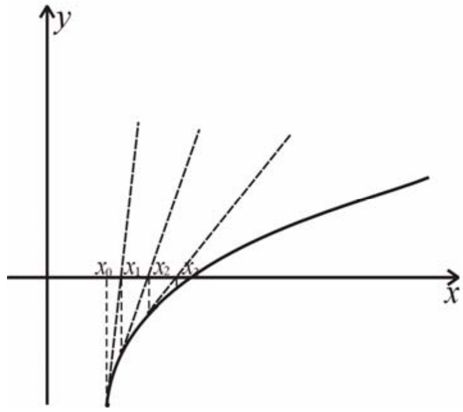


Рисунок3 – Приближение к корню уравнения методом касательных

1.4 Метод Чебышева третьего порядка

Производные функции $f(x)$ первого, второго и третьего порядков непрерывны на отрезке $[a,b]$, причем $f'(x) \ne 0$ на $[a,b]$. Пусть начальное приближение $x_0 \in [a,b]$, тогда итерационный процесс осуществляется по формуле

$$x_n = x_{n-1} - \frac{f(x_{n-1})}{f'(x_{n-1})} - \frac{f''(x_{n-1}) \cdot f^2(x_{n-1})}{2 \cdot (f'(x_{n-1}))^3}, \quad n = 1,2,... \qquad (4)$$

Данный метод обладает скоростью сходимости порядка 3, оценка погрешности имеет вид

$$\left|\bar{x} - x_n\right| \le \left(\alpha \cdot \frac{\beta^3}{3!} \cdot \left|\bar{x} - x_0\right|\right)^{\frac{3^n - 1}{2}},$$

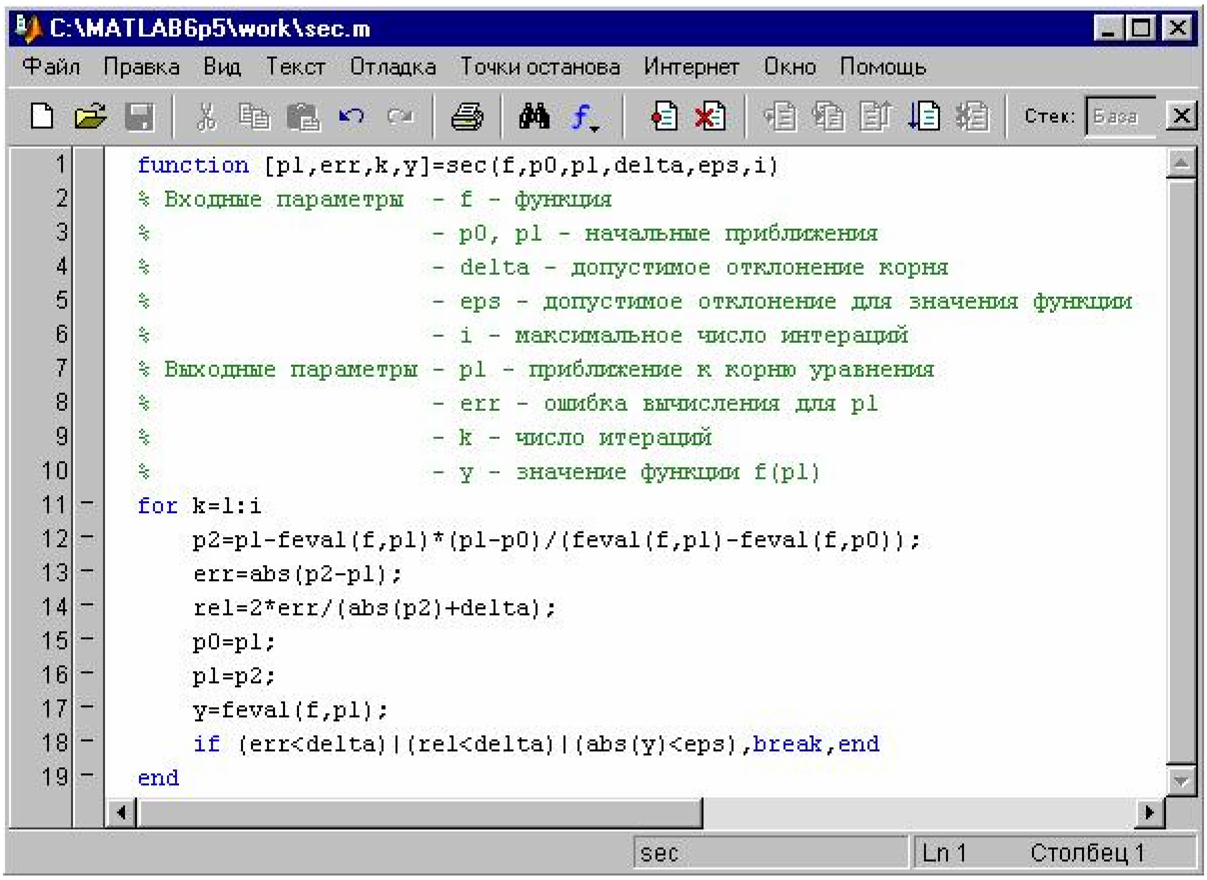
где $\alpha = \max\limits_{x\in[a,b]}\left|F'''(f(x))\right|$, $\beta = \max\limits_{x\in[a,b]}\left|f'(x)\right|$, $F(f(x))$ – функция, обратная $f(x)$ на отрезке $[a,b]$, имеет непрерывные производные первого, второго, и третьего порядков.

Пример 1: Методом секущих найти корень уравнения

$$f(x) = x^3 + 8x^2 - 14x - 20 = 0,$$

расположенного на отрезке $[2,3]$.

Реализация примера с использованием Matlab представлена на рисунке 4.




```
function [p1,err,k,y]=sec(f,p0,p1,delta,eps,i)
% Входные параметры  - f - функция
%                    - p0, p1 - начальные приближения
%                    - delta - допустимое отклонение корня
%                    - eps - допустимое отклонение для значения функции
%                    - i - максимальное число интераций
% Выходные параметры - p1 - приближение к корню уравнения
%                    - err - ошибка вычисления для p1
%                    - k - число итераций
%                    - y - значение функции f(p1)
for k=1:i
    p2=p1-feval(f,p1)*(p1-p0)/(feval(f,p1)-feval(f,p0));
    err=abs(p2-p1);
    rel=2*err/(abs(p2)+delta);
    p0=p1;
    p1=p2;
    y=feval(f,p1);
    if (err<delta)|(rel<delta)|(abs(y)<eps),break,end
end
```

Рисунок 4 – Листинг программы «Метод секущих»

Результат работы программы «Метод секущих»:

```
>> [p1,err,k,y]=sec('f',2,3,10e-4,10e-5,10)
p1  = 2.23944936855661
err = 3.852721542894066e-004
k   = 4
y   =-8.587285155670088e-005
>>
```

Для нахождения корней полинома в пакете Matlab предусмотрена соответствующая функция **roots()**, возвращающая вектор-столбец, компоненты которого являются корнями полинома.

Пример 2: Найти корень уравнения $x^3 + 8x^2 - 14x - 20 = 0$, используя функцию roots().

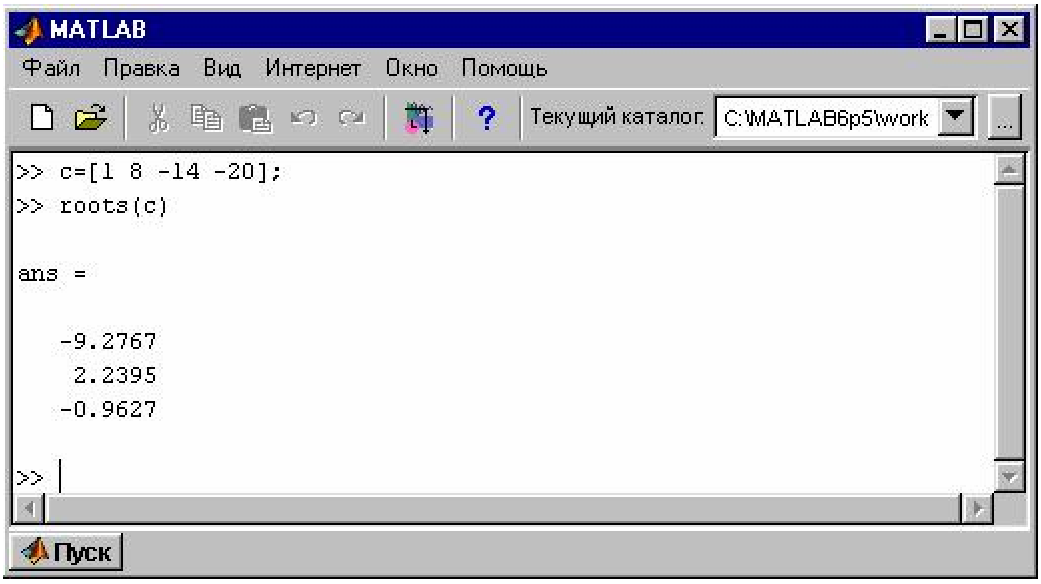


Рисунок 5 – Пример использования функции roots().

Контрольные вопросы

1. Поясните геометрический смысл метода половинного деления.

2. Назовите основную сущность итерационных методов.

3. Почему в методе касательных начальное приближение $x_0 \in [a,b]$ целесообразно выбирать из условия $f(x_0) \cdot f''(x_0) > 0$?

4. Чему равны порядки сходимости рассмотренных методов?

5. Поясните, почему метод касательных является частным случаем метода Чебышева?

Индивидуальные задания

1. Найти корень уравнения $f(x) = 0$ на отрезке $[a,b]$ с точностью $\varepsilon = 10^{-4}$ методами касательных и Чебышева. Определить число итераций и оценить погрешности решения.

2. Реализовать алгоритмы указанных методов и выполнить проверку, используя возможности пакета Matlab.

Номер варианта	Уравнение $f(x)=0$	Отрезок $[a,b]$
1	$x^3+0.3\cdot x^2+1.8\cdot x-15=0$	$[2,3]$
2	$x^3+0.2\cdot x^2+5\cdot x+4=0$	$[-1,0]$
3	$x^3+0.7\cdot x^2+4.7\cdot x-1.5=0$	$[0,2]$
4	$x^3+11\cdot x^2-8\cdot x-25=0$	$[1.5,3]$
5	$x^3+3\cdot x^2+3\cdot x+4=0$	$[-3,-2]$
6	$x^3-2\cdot x-0.4=0$	$[1,2]$
7	$x^3+x^2+3\cdot x-13=0$	$[1,3]$
8	$x^3+x^2+3.8\cdot x-2=0$	$[0,1]$
9	$x^3+3.5x^2+12\cdot x+19=0$	$[-3,-1.5]$
10	$x^3+6x^2-17\cdot x-21=0$	$[2,3.5]$

§ 2 Решение систем линейных уравнений

Пусть дана система n линейных алгебраических уравнений с n неизвестными.

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = a_{1,n+1}, \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = a_{2,n+1}, \\ \cdot \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n = a_{n,n+1}. \end{cases} \tag{5}$$

2.1 Метод Гаусса с выбором главного элемента

Расширенная прямоугольная матрица коэффициентов системы (5) имеет вид

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1q} & \cdots & a_{1n} & a_{1,n+1} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{p1} & \cdots & a_{pq} & \cdots & a_{pn} & a_{p,n+1} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{n1} & \cdots & a_{nq} & \cdots & a_{nn} & a_{n,n+1} \end{pmatrix}.$$

Среди элементов матрицы a_{ij}, $\left(i, j = \overline{1,n}\right)$ выберем главный элемент – наибольший по модулю, например a_{pq}. Строка с номером p – главная строка.

Вычисляем множители

$$m_i = -\frac{a_{iq}}{a_{pq}},\ i \neq p.$$

Преобразуем матрицу A следующим образом: Складываем почленно каждую неглавную i-ю строку с главной строкой, умноженной на m_i. В результате получим матрицу, у которой все элементы q-го столбца, кроме a_{pq} равны нулю. Отбрасывая этот столбец и главную строку, получим новую матрицу A_1 размером $n \times (n-1)$. Над матрицей A_1 проделываем те же операции и получаем матрицу A_2, и т. д. После $(n-1)$-го шага объединяем все главные строки, получаем треугольную матрицу вида (6).

Из последнего уравнения определяется x_n, а затем последовательно остальные значения неизвестных x_i, $i=\overline{n-1,1}$.

$$A_{n-1}=\begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} & a_{1,n+1} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2n}^{(1)} & a_{2,n+1}^{(1)} \\ 0 & 0 & a_{33}^{(2)} & \cdots & a_{3n}^{(2)} & a_{3,n+1}^{(2)} \\ . & . & . & . & . & . \\ 0 & 0 & 0 & \cdots & a_{nn}^{(n-1)} & a_{n,n+1}^{(n-1)} \end{pmatrix}. \qquad (6)$$

2.2 Метод простой итерации

Запишем систему (5) в следующем виде:

$$\begin{cases} x_1=\alpha_{11}x_1+\alpha_{12}x_2+\ldots+\alpha_{1n}x_n+a_{1,n+1}, \\ x_2=\alpha_{21}x_1+\alpha_{22}x_2+\ldots+\alpha_{2n}x_n+a_{2,n+1}, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ x_n=\alpha_{n1}x_1+\alpha_{n2}x_2+\ldots+\alpha_{nn}x_n+a_{n,n+1}. \end{cases} \qquad (7)$$

Исходя из произвольного вектора $x^{(0)}$

$$x^{(0)}=\begin{pmatrix} x_1^{(0)} \\ x_2^{(0)} \\ \cdots \\ x_n^{(0)} \end{pmatrix},$$

строим итерационный процесс

$$\begin{cases} x_1^{(k)}=\alpha_{11}x_1^{(k-1)}+\alpha_{12}x_2^{(k-1)}+\ldots+\alpha_{1n}x_n^{(k-1)}+a_{1,n+1}, \\ x_2^{(k)}=\alpha_{21}x_1^{(k-1)}+\alpha_{22}x_2^{(k-1)}+\ldots+\alpha_{2n}x_n^{(k-1)}+a_{2,n+1}, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ x_n^{(k)}=\alpha_{n1}x_1^{(k-1)}+\alpha_{n2}x_2^{(k-1)}+\ldots+\alpha_{nn}x_n^{(k-1)}+a_{n,n+1}. \end{cases} \qquad (8)$$

Итерационная последовательность точек n-мерного пространства

$$x^{(0)},\ x^{(1)},\ \ldots,\ x^{(k)},\ \ldots$$

сходится и ее предел является решением системы (7) при выполнении одного из условий

$$\sum_{j=1}^{n}\left|\alpha_{ij}\right|\le\lambda<1,\ \ i=\overline{1,n} \qquad (9)$$

или

$$\sum_{i=1}^{n}\left|\alpha_{ij}\right| \le \gamma < 1, \quad j=\overline{1,n}. \tag{10}$$

Оценка погрешности приближенного решения $x^{(k)}$ может быть вычислена с помощью формул

$$\left|x_i - x_i^{(k)}\right| \le \frac{\lambda}{1-\lambda}\max_{j=\overline{1,n}}\left|x_j^{(k)} - x_j^{(k-1)}\right|$$

или

$$\left|x_i - x_i^{(k)}\right| \le \frac{\gamma}{1-\gamma}\sum_{j=1}^{n}\left|x_j^{(k)} - x_j^{(k-1)}\right|.$$

Для установления момента прекращения итераций при достижении заданной точности ε может быть использована формула

$$\left|x_i^{(k)} - x_i^{(k-1)}\right| \le \varepsilon \cdot \frac{\lambda}{1-\lambda}.$$

2.3 Метод Зейделя

Данный метод является модификацией метода простой итерации. Основная идея состоит в том, что на каждом шаге итерационного процесса при вычислении k-го приближения переменной x_i учитываются уже найденные k-е приближения неизвестных $x_1, x_2, \ldots, x_{i-1}$. Таким образом, вычисления производятся по формулам

$$\begin{cases} x_1^{(k)} = \alpha_{11}x_1^{(k-1)} + \alpha_{12}x_2^{(k-1)} + \ldots + \alpha_{1n}x_n^{(k-1)} + a_{1,n+1}, \\ x_2^{(k)} = \alpha_{21}x_1^{(k)} + \alpha_{22}x_2^{(k-1)} + \ldots + \alpha_{2n}x_n^{(k-1)} + a_{2,n+1}, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ x_n^{(k)} = \alpha_{n1}x_1^{(k)} + \alpha_{n2}x_2^{(k)} + \ldots + \alpha_{n,n-1}x_{n-1}^{(k)} + \alpha_{nn}x_n^{(k-1)} + a_{n,n+1}. \end{cases}$$

Достаточные условия сходимости (9), (10) также являются достаточными условиями сходимости метода Зейделя.

Существует возможность автоматического преобразования исходной системы к виду, обеспечивающему сходимость итерационного процесса метода Зейделя. Для этого систему (5) представим в векторно-матричной форме

$$\overline{A} \cdot x = b \tag{13}$$

и умножим левую и правую части уравнения (13) на $\overline{A}^T$, получим равносильную систему

$$C \cdot x = D, \quad C = \overline{A}^T \cdot \overline{A}, \;\; D = \overline{A}^T \cdot b. \tag{14}$$

Систему уравнений (14) называется нормальной и обладает свойствами:

– матрица C – симметрическая;

– все элементы главной диагонали матрицы C положительны.

Указанные свойства позволяют приводить систему (14) к виду, пригодному для итерационного процесса Зейделя:

$$x_i = \sum_{i \neq j} \alpha_{i,j} x_j + \beta_j, \;\; i = \overline{1,n},$$

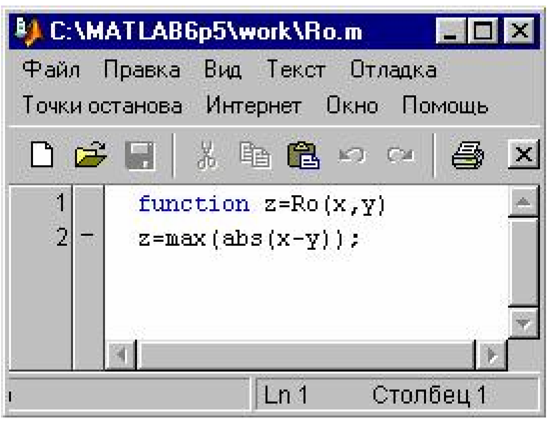
$$\alpha_{ij} = -\frac{c_{ij}}{c_{ii}}, i \neq j, \qquad \beta_i = \frac{d_i}{c_{ii}}, \; i = \overline{1,n}.$$

Пример 3: Методом простой итерации найти решение системы уравнений

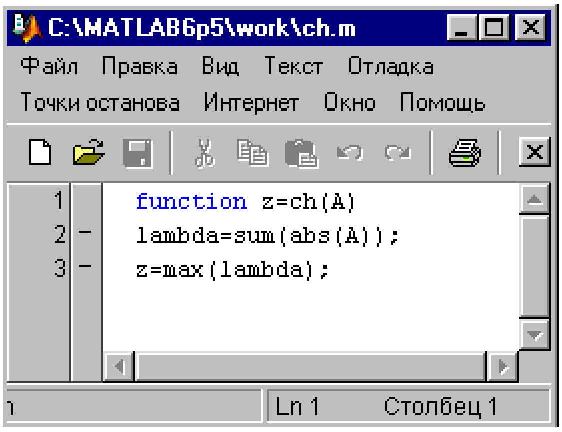
$$\begin{cases} x_1 = 0.12x_1 - 0.23x_2 + 0.25x_3 - 0.16x_4 + 1.24, \\ x_2 = 0.14x_1 + 0.34x_2 - 0.18x_3 + 0.24x_4 - 0.89, \\ x_3 = 0.13x_1 + 0.03x_2 + 0.46x_3 - 0.32x_4 + 1.15, \\ x_4 = 0.12x_1 - 0.05x_2 + 0.15x_4 - 0.57, \end{cases} \tag{11}$$

с точность $\varepsilon = 10^{-3}$.

1. Функция Ro возвращает значение расстояния между точками в пространстве с метрикой $\rho(x, y) = \max\limits_{i=\overline{1,n}} |x_i - y_i|$.



2. Функция Ch возвращает максимальное значение λ из формулы (9)



3. Функция Iter возвращает решение системы линейных уравнений полученное с заданной точностью ε (рисунок 6).

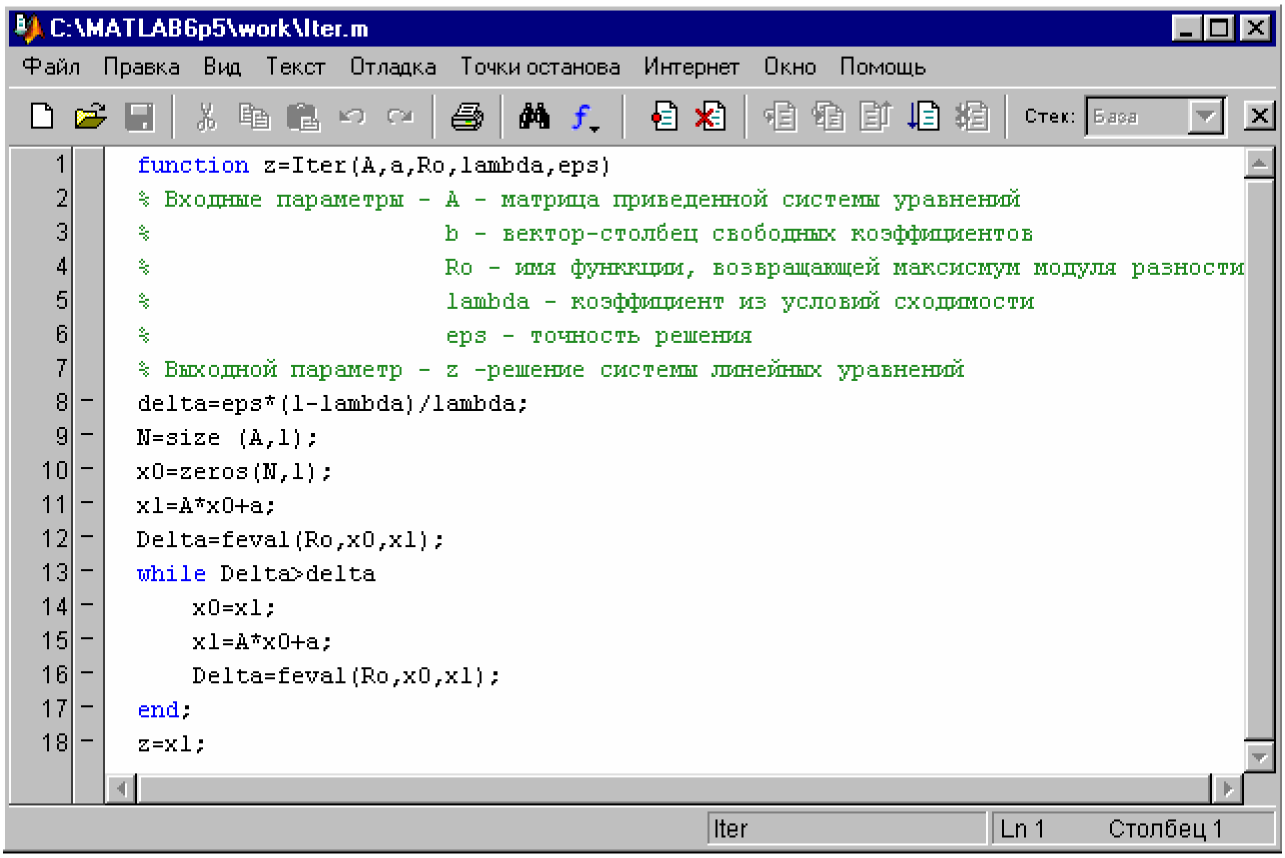


Рисунок 6 – Листинг программы «Метод простой итерации для решения систем линейных уравнений»

Результат работы программы, определяющей решение системы линейных уравнений (11) методом простой итерации

```
>> A=[0.12,-0.23,0.25,-0.16;0.14,0.34,-0.18,0.24;0.13,0.03,0.46,-0.32;0.12,-0.05,0,0.15]

A = 0.1200   -0.2300    0.2500   -0.1600
    0.1400    0.3400   -0.1800    0.2400
    0.1300    0.0300    0.4600   -0.3200
    0.1200   -0.0500         0    0.1500

>> b=[1.24;-0.89;1.15;-0.57]

b =  1.2400
    -0.8900
     1.1500
    -0.5700

>> lambda=ch(A)

 lambda = 0.8900

>> x=Iter(A,b,'Ro',lambda,10e-3)

 x = 2.6633
    -1.6192
     2.7984
    -0.1995
```

Для решения систем линейных уравнений средствами пакета Matlab применяются операторы возведения в степень и умножения, действие которых определяется правилами линейной алгебры.

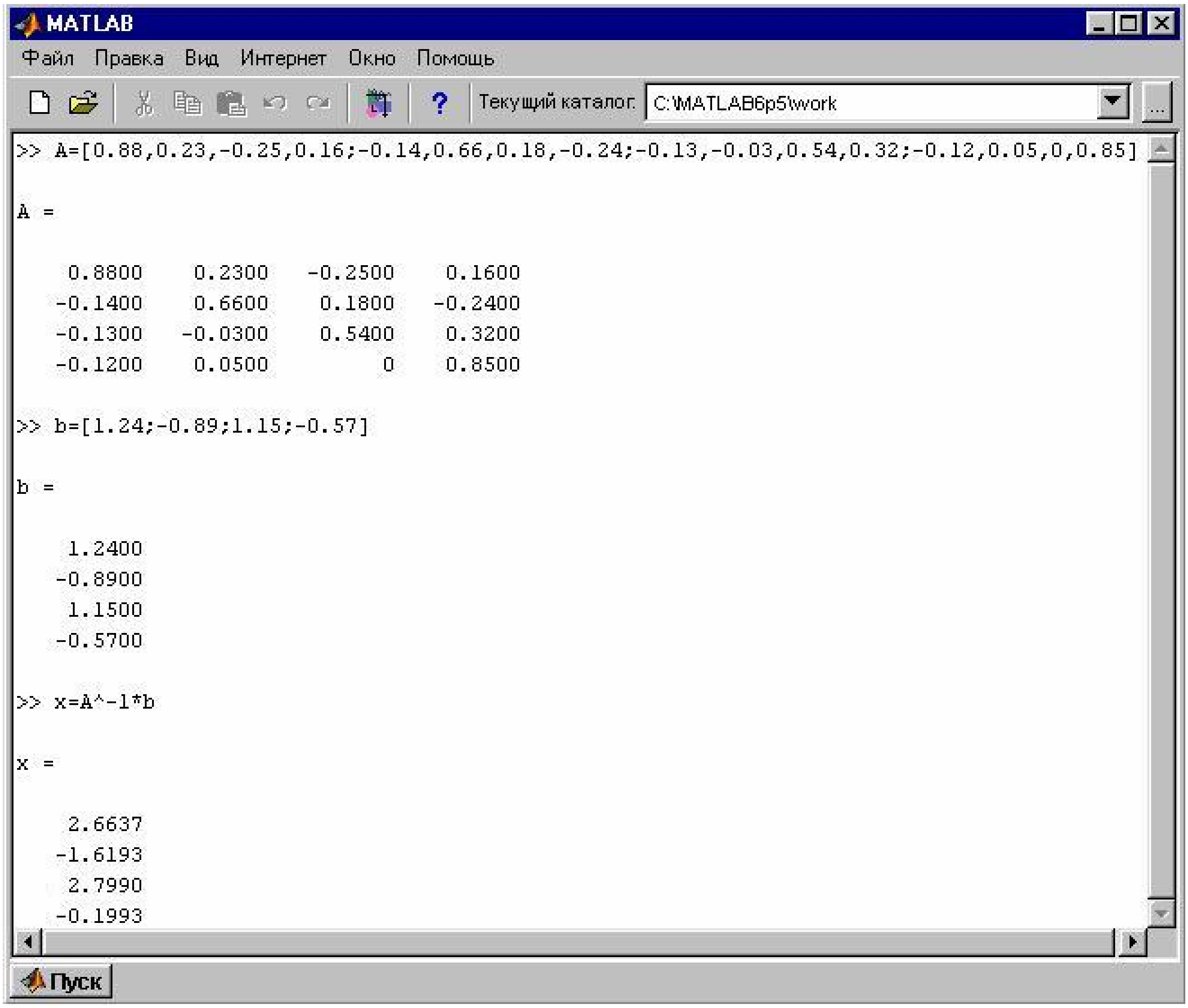
Пример 4: Приведем систему (11) к обычному виду

$$\begin{cases} x_1 - 0.12x_1 + 0.23x_2 - 0.25x_3 + 0.16x_4 = 1.24, \\ x_2 - 0.14x_1 - 0.34x_2 + 0.18x_3 - 0.24x_4 = -0.89, \\ x_3 - 0.13x_1 - 0.03x_2 - 0.46x_3 + 0.32x_4 = 1.15, \\ x_4 - 0.12x_1 + 0.05x_2 - 0.15x_4 = -0.57, \end{cases}$$

или

$$\begin{cases} 0.88x_1 + 0.23x_2 - 0.25x_3 + 0.16x_4 = 1.24, \\ -0.14x_1 + 0.66x_2 + 0.18x_3 - 0.24x_4 = -0.89, \\ -0.13x_1 - 0.03x_2 + 0.54x_3 + 0.32x_4 = 1.15, \\ -0.12x_1 + 0.05x_2 + 0.85x_4 = -0.57, \end{cases} \tag{12}$$

Решение системы (12) средствами пакета Matlab может быть записано следующим образом:



Контрольные вопросы

1. Какие существуют основные схемы, реализующие метод Гаусса?

2. В чем заключается основное преимущество метода Гаусса с выбором главного элемента?

3. Какие существуют способы приведения исходной матрицы к виду, удобному для решения методом простой итерации?

4. Каким образом может быть выбран вектор начальных приближений?

5. Назовите достаточное условие сходимости метода простой итерации.

6. В чем заключается преимущество метода Зейделя при программировании?

Индивидуальные задания

1. Реализуйте средствами пакета Matlab метод Гаусса с выбором главного элемента и решите систему уравнений (13). Значения матрицы $\overline{A}$ и вектора b приведены в таблице.

2. Приведите систему уравнений (13) к виду, пригодному для итерационного процесса Зейделя, реализуйте метод в пакете Matlab и решите систему уравнений с точность $\varepsilon = 10^{-3}$ и исходными данными, приведенными в таблице.

Номер варианта	Значения матрицы $\overline{A}$ и вектора b
1	$\overline{A} = \begin{pmatrix} 2.1 & -4.5 & -2 \\ 3 & 2.5 & 4.3 \\ -6 & 3.5 & 2.5 \end{pmatrix}, b = \begin{pmatrix} 19.07 \\ 3.21 \\ -18.25 \end{pmatrix}$
2	$\overline{A} = \begin{pmatrix} -6.7 & 5.9 & 4.1 & -9.1 \\ 5.4 & -8.7 & 11.2 & -0.5 \\ 2.7 & -4.9 & 6.8 & 3.3 \\ 1.6 & 7.2 & 8.3 & -7.9 \end{pmatrix}, b = \begin{pmatrix} 7.12 \\ 58.07 \\ 47.1 \\ 36.8 \end{pmatrix}$
3	$\overline{A} = \begin{pmatrix} 3.81 & 0.25 & 1.28 & 0.75 \\ 2.25 & 1.32 & 4.58 & 0.49 \\ 5.31 & 6.28 & 0.98 & 1.04 \\ 9.39 & 2.45 & 3.35 & 2.28 \end{pmatrix}, b = \begin{pmatrix} 4.21 \\ 6.47 \\ 2.38 \\ 10.48 \end{pmatrix}$
4	$\overline{A} = \begin{pmatrix} 2.34 & -4.21 & -11.61 \\ 8.04 & 5.22 & 0.27 \\ 3.92 & -7.99 & 8.37 \end{pmatrix}, b = \begin{pmatrix} 14.41 \\ -6.44 \\ 55.56 \end{pmatrix}$

5	$\overline{A}=\begin{pmatrix}0.18 & 0.25 & -0.44\\ 0.42 & -0.35 & 1.12\\ 1.14 & 0.12 & -0.83\end{pmatrix}, b=\begin{pmatrix}1.15\\ 0.86\\ 0.68\end{pmatrix}$
6	$\overline{A}=\begin{pmatrix}21.54 & -95.5 & -96.12\\ 10.22 & -91.06 & -7.34\\ 51.21 & 12.29 & 86.45\end{pmatrix}, b=\begin{pmatrix}-49.93\\ -12.46\\ 60.81\end{pmatrix}$
7	$\overline{A}=\begin{pmatrix}2.51 & -3.12 & 4.64\\ -3.25 & 2.62 & 1.85\\ -6.53 & -3.5 & 7.31\end{pmatrix}, b=\begin{pmatrix}-1.05\\ -14.46\\ -17.73\end{pmatrix}$
8	$\overline{A}=\begin{pmatrix}8.2 & 1.4 & -2.3 & 0.2\\ -1.6 & 5.4 & -7.7 & 3.1\\ 0.7 & 1.9 & -8.5 & 4.8\\ 5.3 & -5.9 & 2.7 & -7.9\end{pmatrix}, b=\begin{pmatrix}32.76\\ 54.39\\ 59.18\\ -71.95\end{pmatrix}$
9	$\overline{A}=\begin{pmatrix}4.21 & 22.42 & 3.85\\ 2.31 & 31.49 & 1.52\\ 3.49 & 4.85 & 28.72\end{pmatrix}, b=\begin{pmatrix}30.24\\ 4.095\\ 42.81\end{pmatrix}$
10	$\overline{A}=\begin{pmatrix}0.63 & 0.07 & 0.2\\ 0.05 & 0.34 & 0.3\\ 0.15 & 0.1 & 0.71\end{pmatrix}, b=\begin{pmatrix}0.34\\ 0.32\\ 0.42\end{pmatrix}$

§ 3 Решение систем нелинейных уравнений

Пусть требуется решить систему уравнений

$$\begin{cases} f_1(x_1, x_2, \ldots, x_n) = 0, \\ f_2(x_1, x_2, \ldots, x_n) = 0, \\ . \ . \ . \ . \ . \ . \ . \ . \ . \ . \ . \ . \\ f_n(x_1, x_2, \ldots, x_n) = 0, \end{cases} \qquad (15)$$

где f_1, f_2, …, f_n – заданные нелинейные вещественнозначные функции n вещественных переменных $x_1,\ x_2, \ldots, x_n$.

Введем n-мерные векторы

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{pmatrix}, \quad f(x) = \begin{pmatrix} f_1(x) \\ f_2(x) \\ \cdots \\ f_n(x) \end{pmatrix}.$$

3.1 Метод Ньютона

Предположим, что найдено q-е приближение

$$x^{(q)} = \left(x_1^{(q)}, x_2^{(q)}, \ldots, x_n^{(q)}\right).$$

Введем в рассмотрение матрицу Якоби системы уравнений (15) относительно переменных $x_1,\ x_2, \ldots, x_n$

$$W(x) = \begin{pmatrix} \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial x_2} & \cdots & \dfrac{\partial f_1}{\partial x_n} \\ \dfrac{\partial f_2}{\partial x_1} & \dfrac{\partial f_2}{\partial x_2} & \cdots & \dfrac{\partial f_2}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \dfrac{\partial f_n}{\partial x_1} & \dfrac{\partial f_n}{\partial x_2} & \cdots & \dfrac{\partial f_n}{\partial x_n} \end{pmatrix}.$$

Тогда последовательные приближения находятся по формуле

$$x^{(q+1)} = x^{(q)} - W^{-1}\left(x^{(q)}\right) \cdot f\left(x^{(q)}\right), \ q = 0, 1, 2, \ldots$$

3.2 Метод скорейшего спуска

В этом методе решение системы (15) сводится к задаче отыскания минимумов функции

$$\Phi(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n} f_i^2(x_1, x_2, \ldots, x_n). \quad (16)$$

Так как функция (16) неотрицательная, то найдется точка $x^* = (x_1^*, x_2^*, \ldots, x_n^*)$ такая, что

$$\Phi(x_1, x_2, \ldots, x_n) \geq \Phi(x_1^*, x_2^*, \ldots, x_n^*) \geq 0, \quad \forall (x_1, x_2, \ldots, x_n) \in R^n.$$

Следовательно, если удается найти точку x^*, минимизирующую функцию (16) и если при этом

$$\min_{(x_1, x_2, \ldots, x_n) \in R^n} \Phi(x_1, x_2, \ldots, x_n) = \Phi(x_1^*, x_2^*, \ldots, x_n^*) = 0,$$

то точка x^* – истинное решение системы (15).

Последовательность точек $x^{(k)}$ – приближений к точке x^* вычисляется по формуле

$$x^{(k+1)} = x^{(k)} - \lambda_k \cdot \Phi_x\left(x^{(k)}\right),$$

где

$$\Phi_x\left(x^{(k)}\right) = grad\, \Phi\left(x^{(k)}\right) = \left(\frac{\partial \Phi\left(x^{(k)}\right)}{\partial x_1}, \frac{\partial \Phi\left(x^{(k)}\right)}{\partial x_2}, \cdots, \frac{\partial \Phi\left(x^{(k)}\right)}{\partial x_n} \right),$$

а λ_k – находится из условия минимума функции

$$\psi_k(\lambda_k) = \Phi\left(x^{(k)} - \lambda_k \cdot \Phi_x\left(x^{(k)}\right)\right). \quad (17)$$

Рисунок 7 иллюстрирует алгоритм нахождение решения системы уравнений методом наискорейшего спуска.

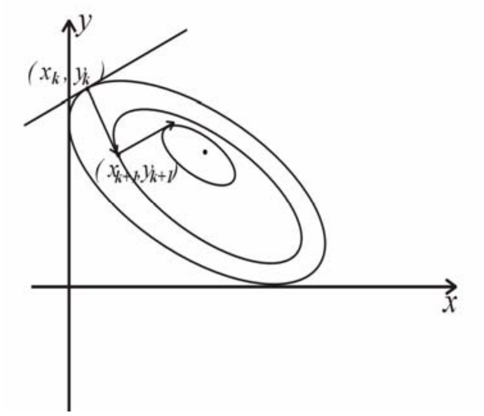


Рисунок 7 – Траектория наискорейшего спуска
для функции (16) двух переменных

Главное достоинство градиентных методов решения нелинейных систем – глобальная сходимость. Процесс градиентного спуска приведет к какой либо точке минимума функции из любой начальной точки. При определенных условиях найденная точка минимума будет искомым решением исходной нелинейной системы уравнений.

Пример 5: Решить систему уравнений методом скорейшего спуска

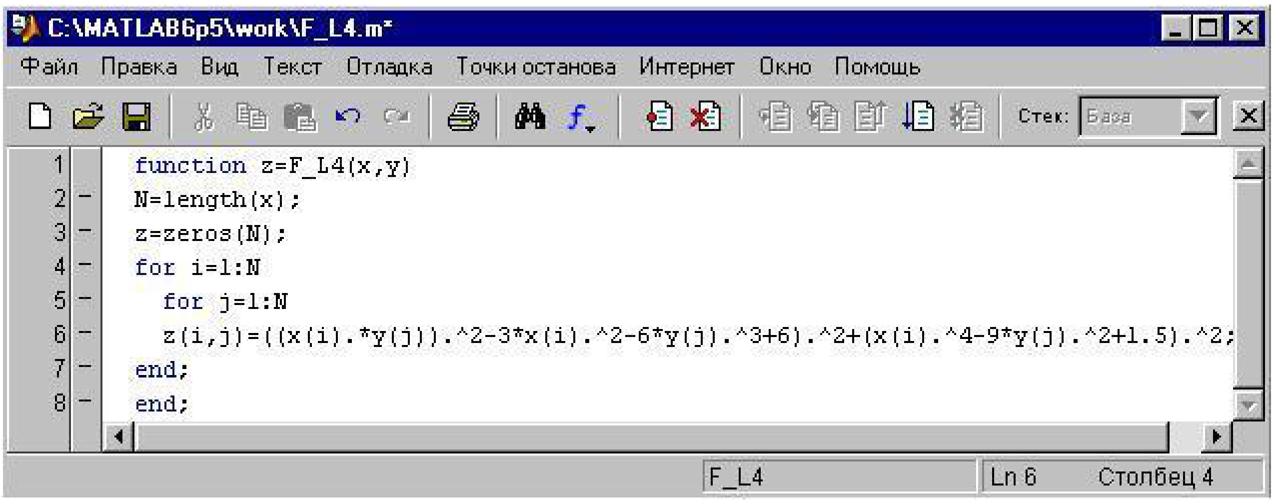
$$\begin{cases} x^2 \cdot y^2 - 3 \cdot x^2 - 6 \cdot y^3 + 6 = 0, \\ x^4 - 9 \cdot y^2 + 1.5 = 0. \end{cases} \tag{18}$$

Для системы (18) запишем явный вид функции (16)

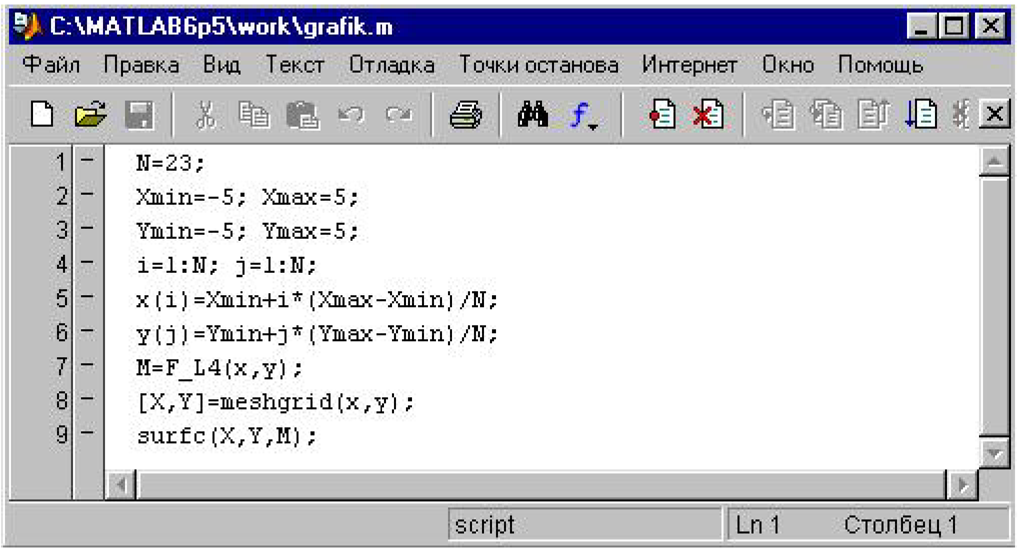
$$\Phi(x, y) = \left(x^2 \cdot y^2 - 3 \cdot x^2 - 6 \cdot y^3 + 6\right)^2 + \left(x^4 - 9 \cdot y^2 + 1.5\right)^2. \tag{19}$$

Реализуем алгоритм поиска экстремума функции (19) в пакете Matlab.

1. Создадим файл, содержащий описание функции, возвращающей значение функции (19) в узлах координатной сетки.



2. Построим график функции (19).



Поверхность (19) и карта линий уровня представлены на рисунке 8.

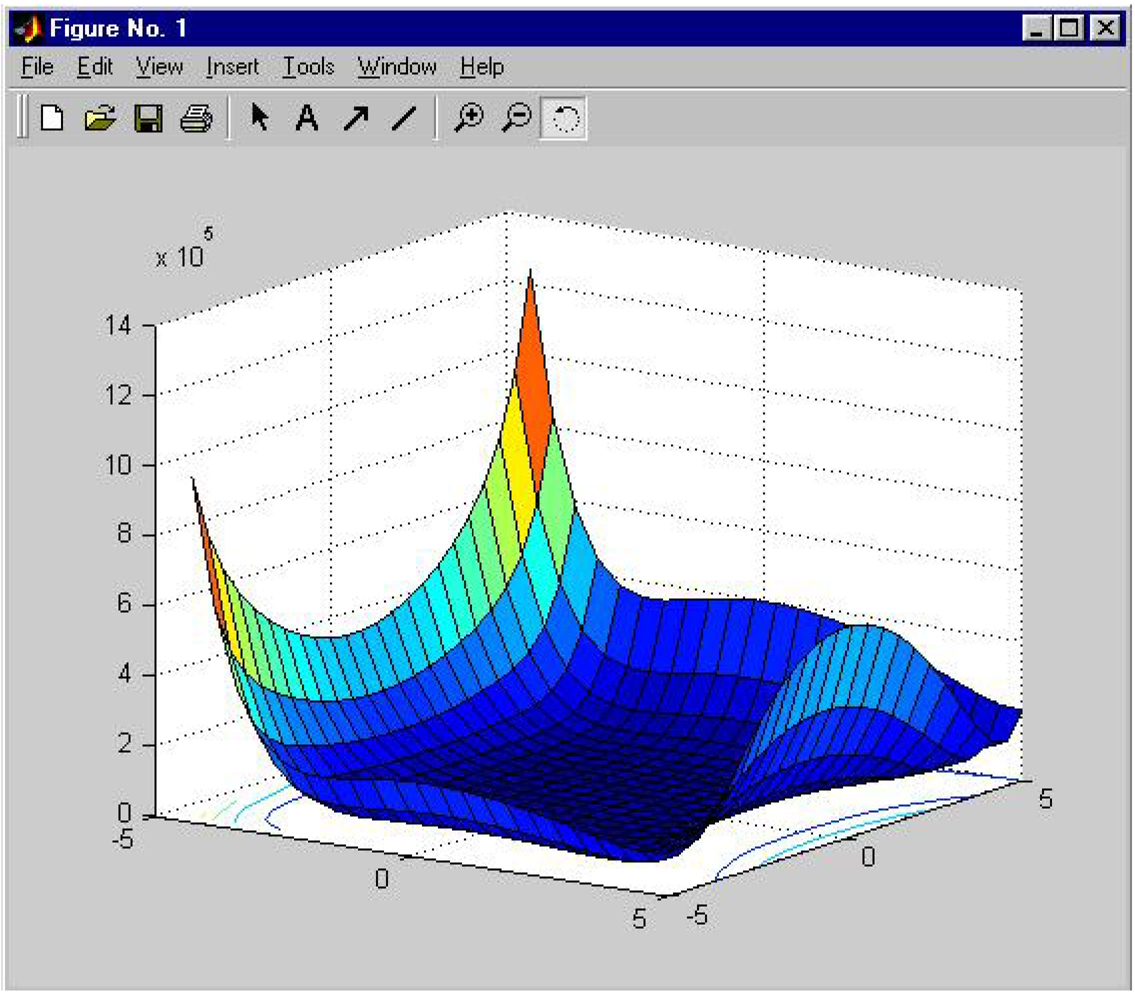


Рисунок 8 – Поверхность, соответствующая функции (19), и ее линии уровня

3. Создадим файлы, содержащие описание функций, возвращающих значения частных производных.

Частную производную по переменной x:

C:\MATLAB6p5\work\Dx_F_L4.m

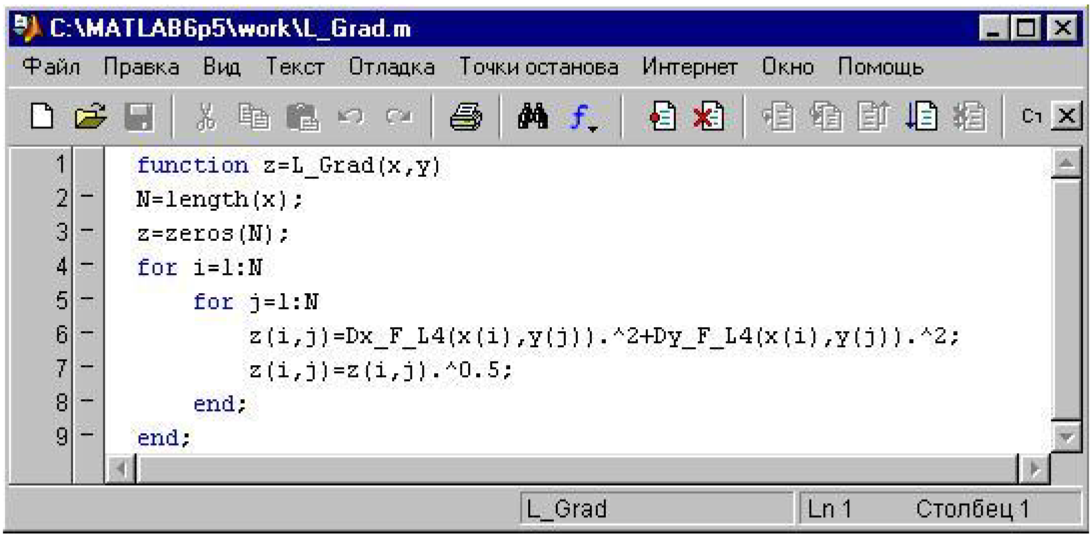
```
function z=Dx_F_L4(x,y)
N=length(x);
z=zeros(N);
i=1:N;
j=1:N;
for i=1:N
 for j=1:N
 z(i,j)=2*((x(i).*y(j)).^2-3*x(i).^2-6*y(j).^3+6).*(2*x(i).*(y(j).^2)-
 -6*x(i))+2*(x(i).^4-9*y(j).^2+1.5).*4*x(i).^3;
end;
end;
```

Частную производную по переменной y:

C:\MATLAB6p5\work\Dy_F_L4.m

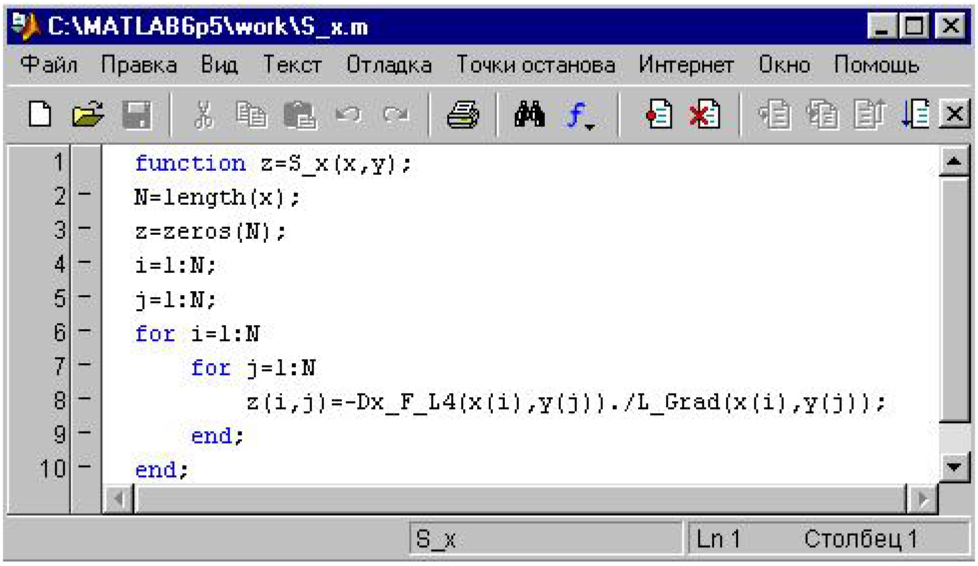
```
function z=Dy_F_L4(x,y)
N=length(x);
z=zeros(N);
i=1:N;
j=1:N;
for i=1:N
 for j=1:N
 z(i,j)=2*((x(i).*y(j)).^2-3*x(i).^2-6*y(j).^3+6).*(2*y(j).*(x(i).^2)-
 -18*y(j).^2)+2*(x(i).^4-9*y(j).^2+1.5).*(-18)*y(j);
end;
end;
```

4. Файл L_Grad.m содержит описание функции, которая возвращает значение длины градиента функции (19).

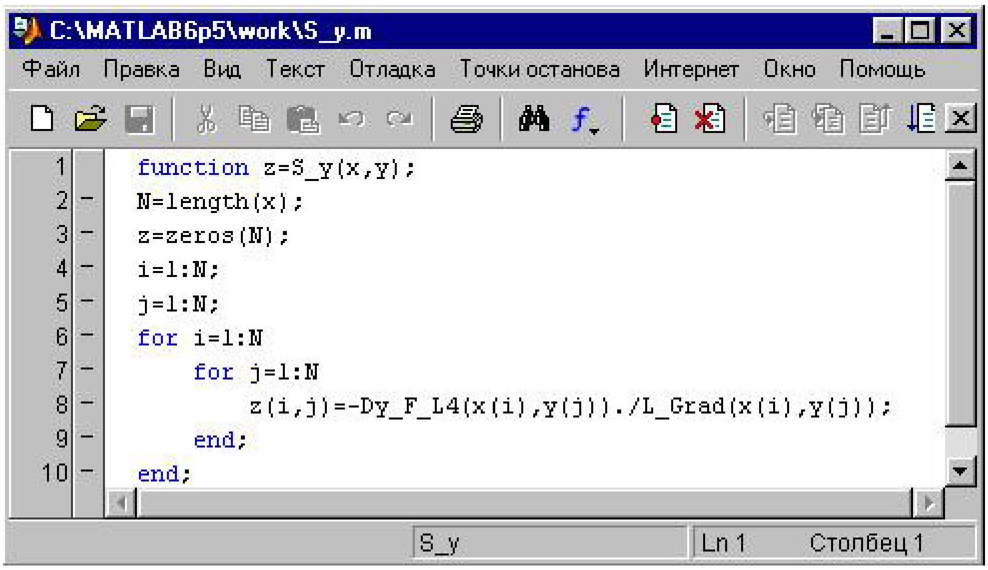



```
function z=L_Grad(x,y)
N=length(x);
z=zeros(N);
for i=1:N
    for j=1:N
        z(i,j)=Dx_F_L4(x(i),y(j)).^2+Dy_F_L4(x(i),y(j)).^2;
        z(i,j)=z(i,j).^0.5;
    end;
end;
```

5. Файлы S_x.m и S_y.m содержат функции, возвращающие координаты нормированного единичного вектора, сонаправленного с вектором, противоположным направлению вектора градиента.

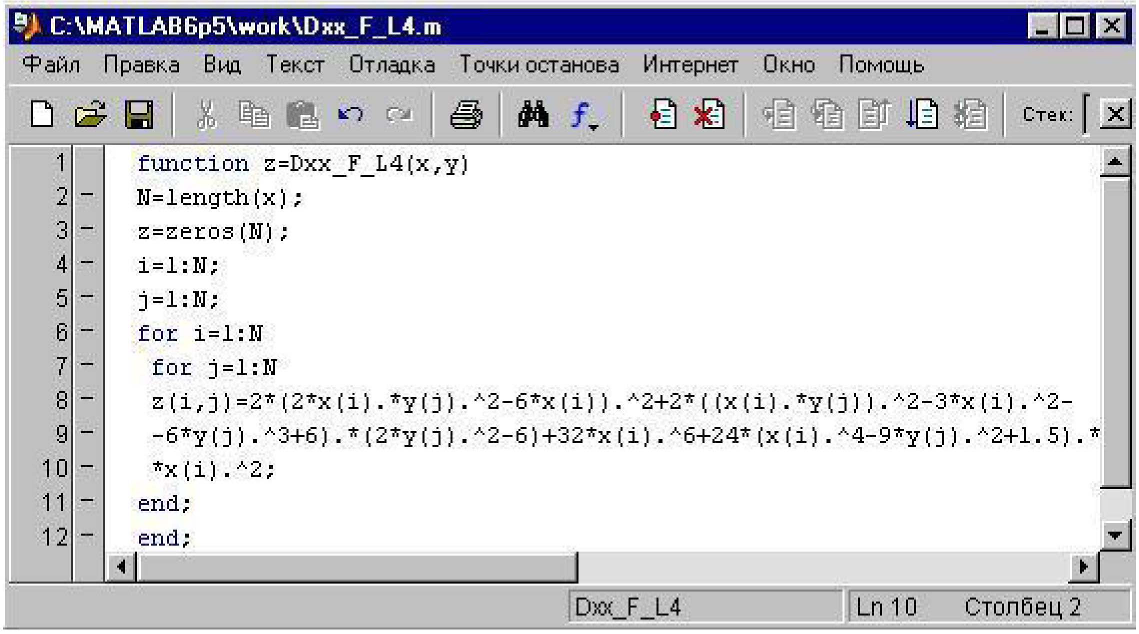


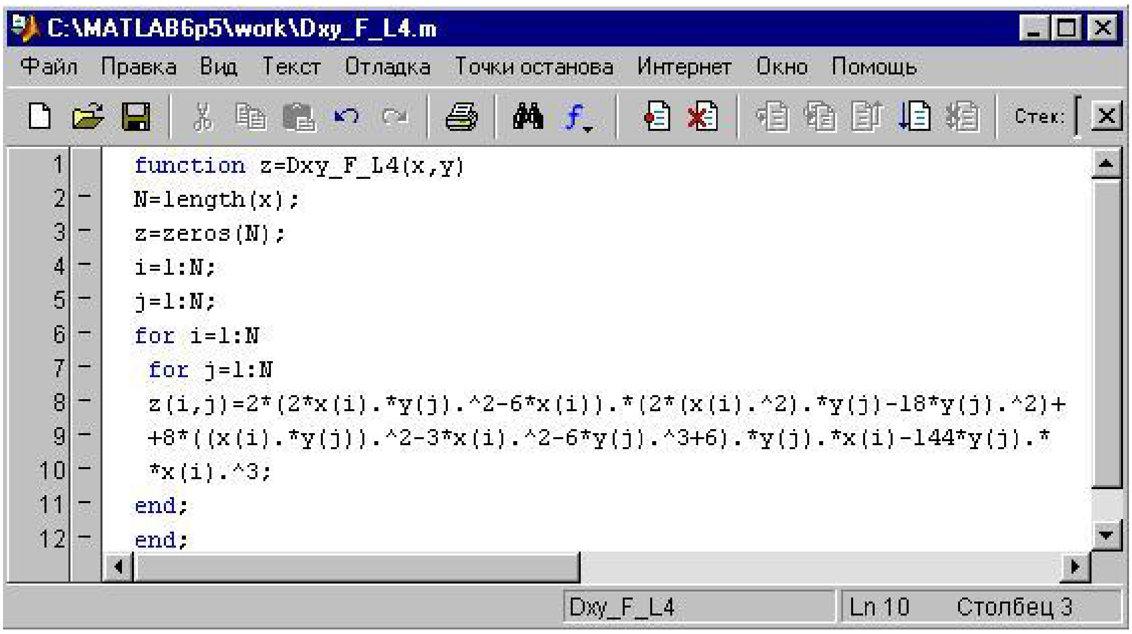

```
function z=S_x(x,y);
N=length(x);
z=zeros(N);
i=1:N;
j=1:N;
for i=1:N
    for j=1:N
        z(i,j)=-Dx_F_L4(x(i),y(j))./L_Grad(x(i),y(j));
    end;
end;
```

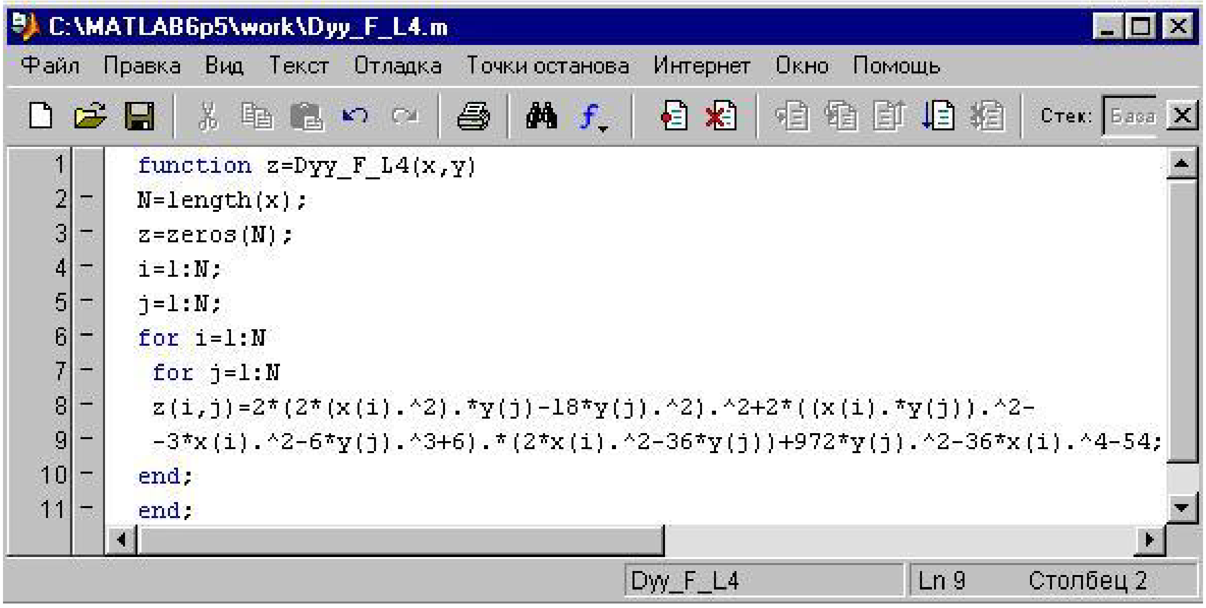



```
function z=S_y(x,y);
N=length(x);
z=zeros(N);
i=1:N;
j=1:N;
for i=1:N
    for j=1:N
        z(i,j)=-Dy_F_L4(x(i),y(j))./L_Grad(x(i),y(j));
    end;
end;
```

6. Создадим файлы Dxx_F_L4.m, Dxy_F_L4.m, Dyy_F_L4.m, которые содержат описание вторых частных производных функции $\Phi(x,y)$: $\dfrac{\partial^2\Phi(x,y)}{\partial x^2}$, $\dfrac{\partial^2\Phi(x,y)}{\partial x\partial y}$, $\dfrac{\partial^2\Phi(x,y)}{\partial y^2}$ соответственно.

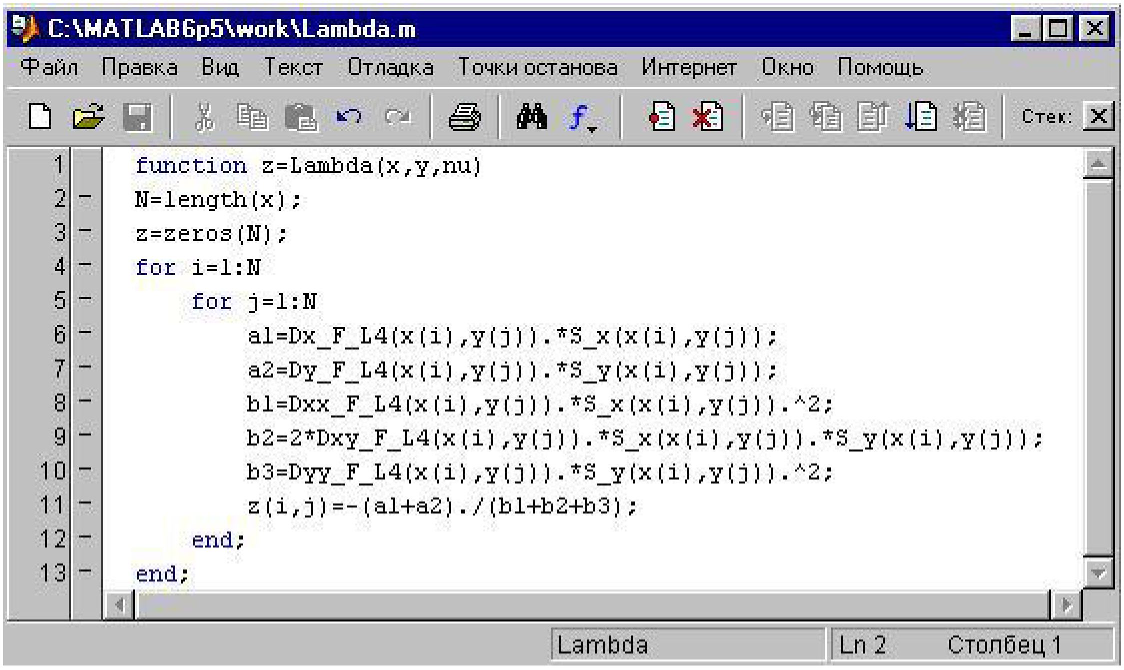







```
function z=Dyy_F_L4(x,y)
N=length(x);
z=zeros(N);
i=1:N;
j=1:N;
for i=1:N
 for j=1:N
 z(i,j)=2*(2*(x(i).^2).*y(j)-18*y(j).^2).^2+2*((x(i).*y(j)).^2-
 -3*x(i).^2-6*y(j).^3+6).*(2*x(i).^2-36*y(j))+972*y(j).^2-36*x(i).^4-54;
end;
end;
```

7. Функция Lambda.m возвращает значения коэффициента λ из уравнения (17).




```
function z=Lambda(x,y,nu)
N=length(x);
z=zeros(N);
for i=1:N
    for j=1:N
        a1=Dx_F_L4(x(i),y(j)).*S_x(x(i),y(j));
        a2=Dy_F_L4(x(i),y(j)).*S_y(x(i),y(j));
        b1=Dxx_F_L4(x(i),y(j)).*S_x(x(i),y(j)).^2;
        b2=2*Dxy_F_L4(x(i),y(j)).*S_x(x(i),y(j)).*S_y(x(i),y(j));
        b3=Dyy_F_L4(x(i),y(j)).*S_y(x(i),y(j)).^2;
        z(i,j)=-(a1+a2)./(b1+b2+b3);
    end;
end;
```

8. Создадим файл MG.m, содержащий описание функции, которая возвращает значения переменных x, y и соответствующее значение функции $\Phi(x, y)$ на каждом шаге итерационного процесса.

C:\MATLAB6p5\work\MG.m

```
function [X,Y,ff]=MG(x0,y0,nu)
X(1)=x0;
Y(1)=y0;
ff(1)=F_L4(x0,y0);
for i=2:nu
    X(i)=X(i-1)+Lambda(X(i-1),Y(i-1),nu)*S_x(X(i-1),Y(i-1));
    Y(i)=Y(i-1)+Lambda(X(i-1),Y(i-1),nu)*S_y(X(i-1),Y(i-1));
    ff(i)=F_L4(X(i),Y(i));
end;
```

9. Найдем решение исходной системы и визуализируем итерационный процесс.

C:\MATLAB6p5\work\rescenie.m

```
x0=2; y0=1;
nu=20;
[X,Y,ff]=MG(x0,y0,nu);
plot(X), grid,
hold on
plot(Y),
hold off
plot(ff),grid,
stairs(X,Y), grid,
plot3(X,Y,ff),grid;
```

Зависимость координат точки решения исходной системы уравнений от номера итерации представлена на графике.

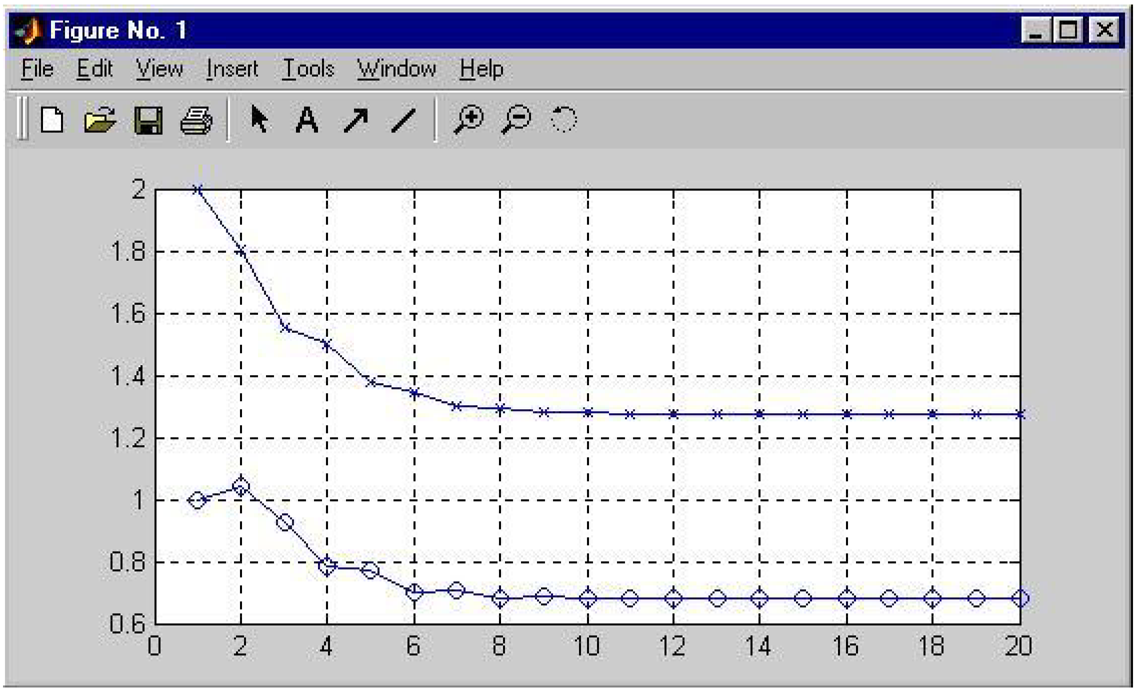
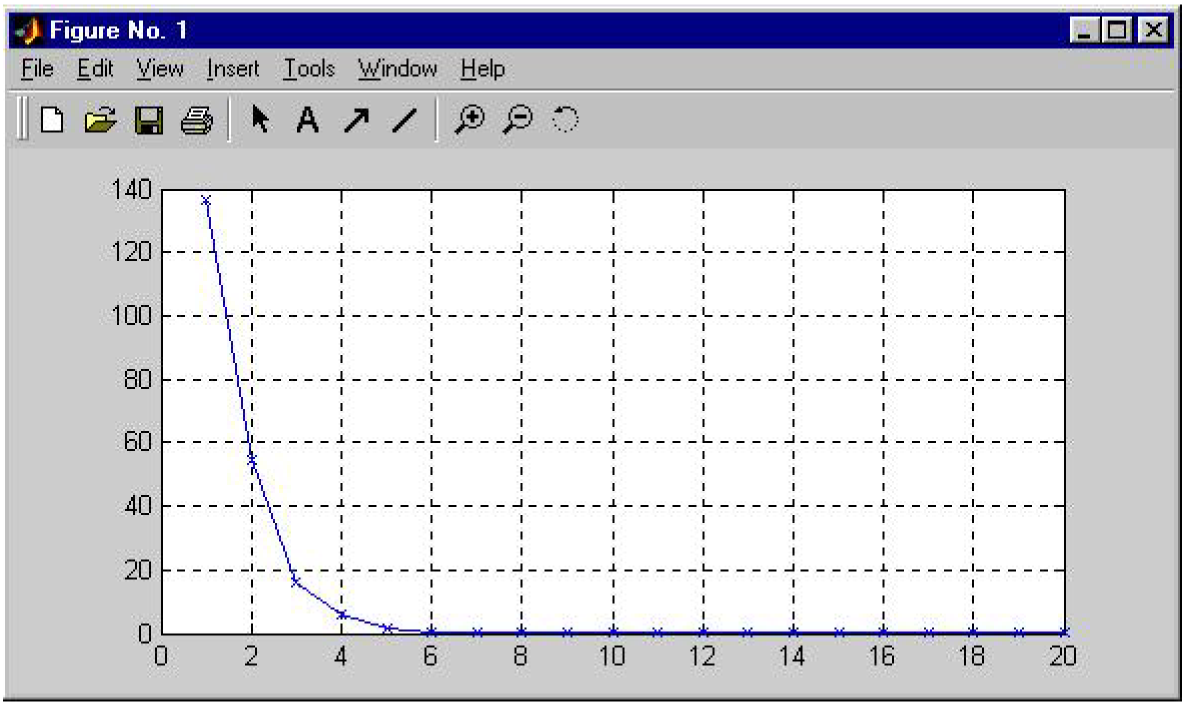


График зависимости значения исследуемой функции (18) от номера итерации изображен на рисунке.



Траектория итерационного процесса в плоскости *X0Y* и в пространстве представлены на соответствующих рисунках.

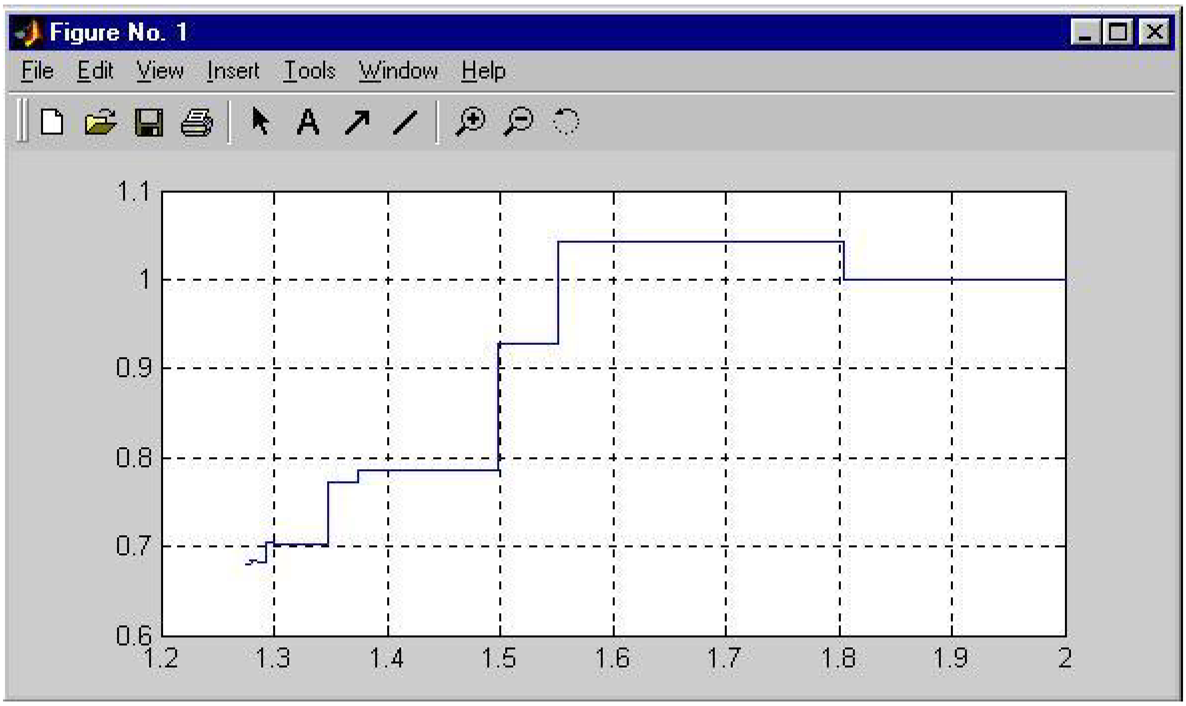


Figure No. 1
File Edit View Insert Tools Window Help
1.1
1
0.9
0.8
0.7
0.6
1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9 2

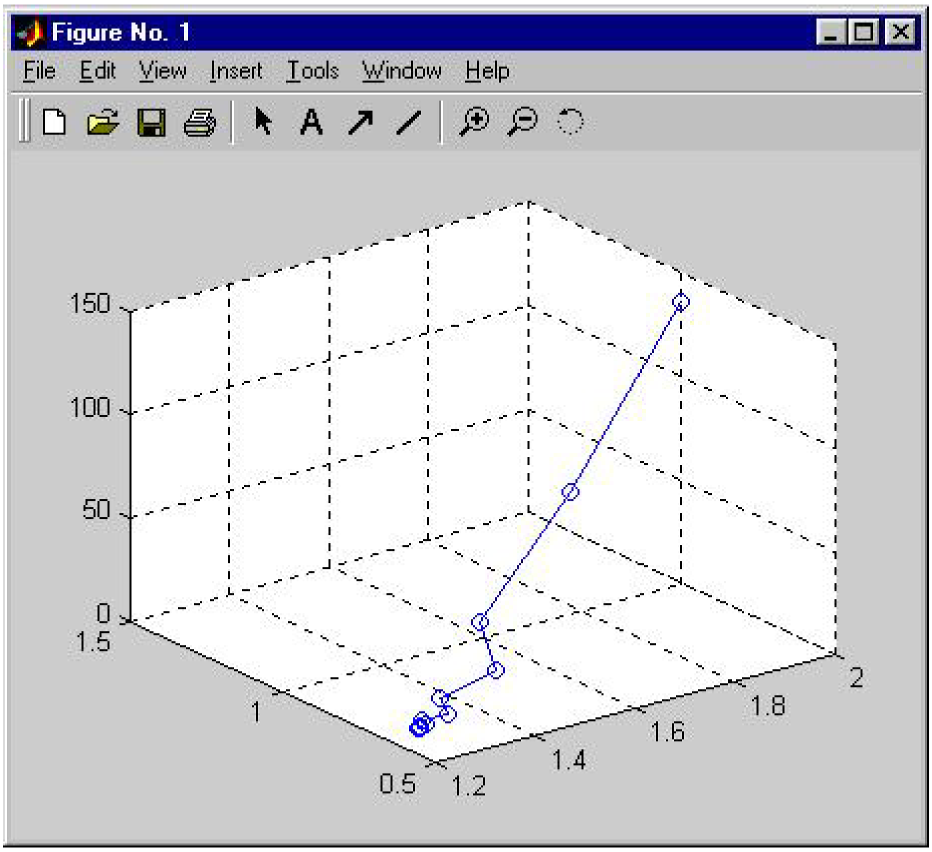
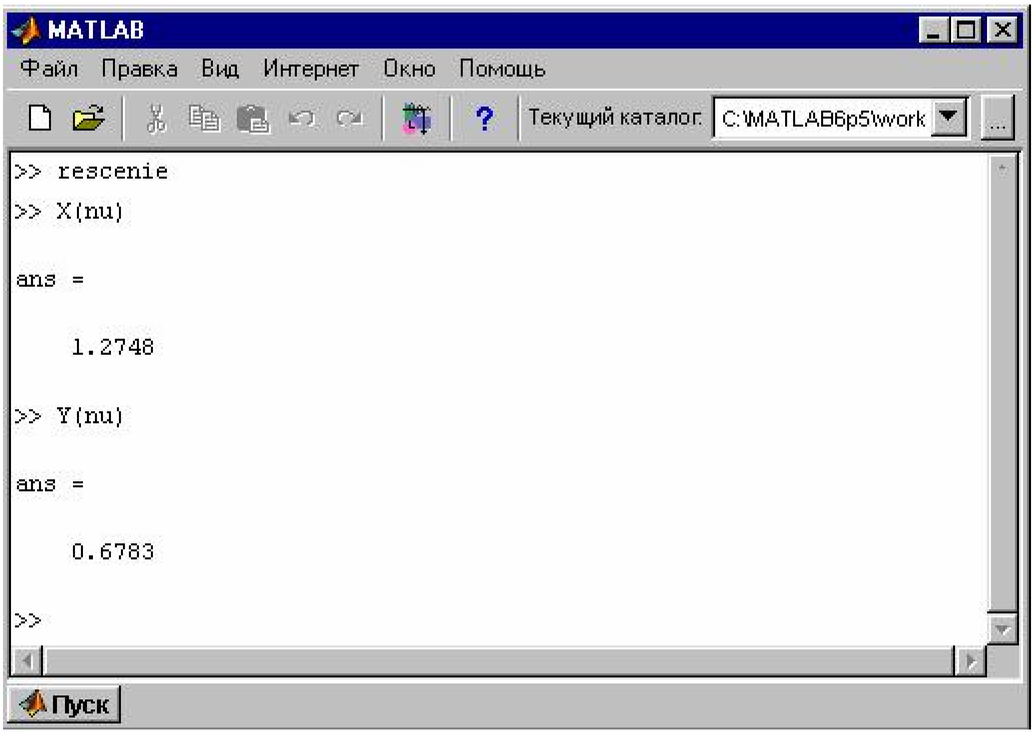


Figure No. 1
File Edit View Insert Tools Window Help
150
100
50
0
1.5
1
0.5
1.2
1.4
1.6
1.8
2

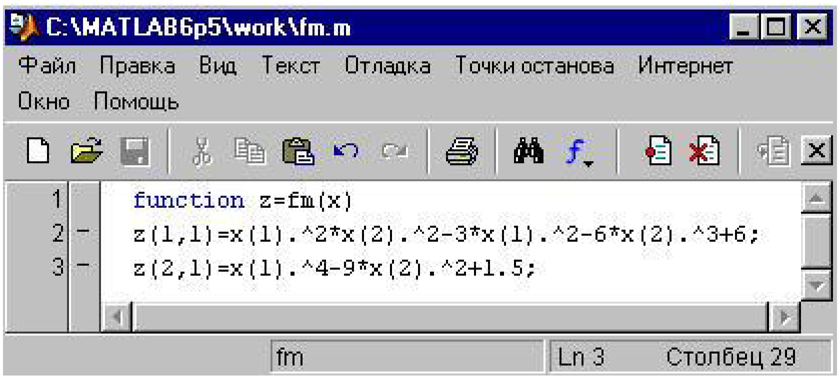
Результат решения системы нелинейных уравнений (18) методом скорейшего спуска приведен в следующем окне:



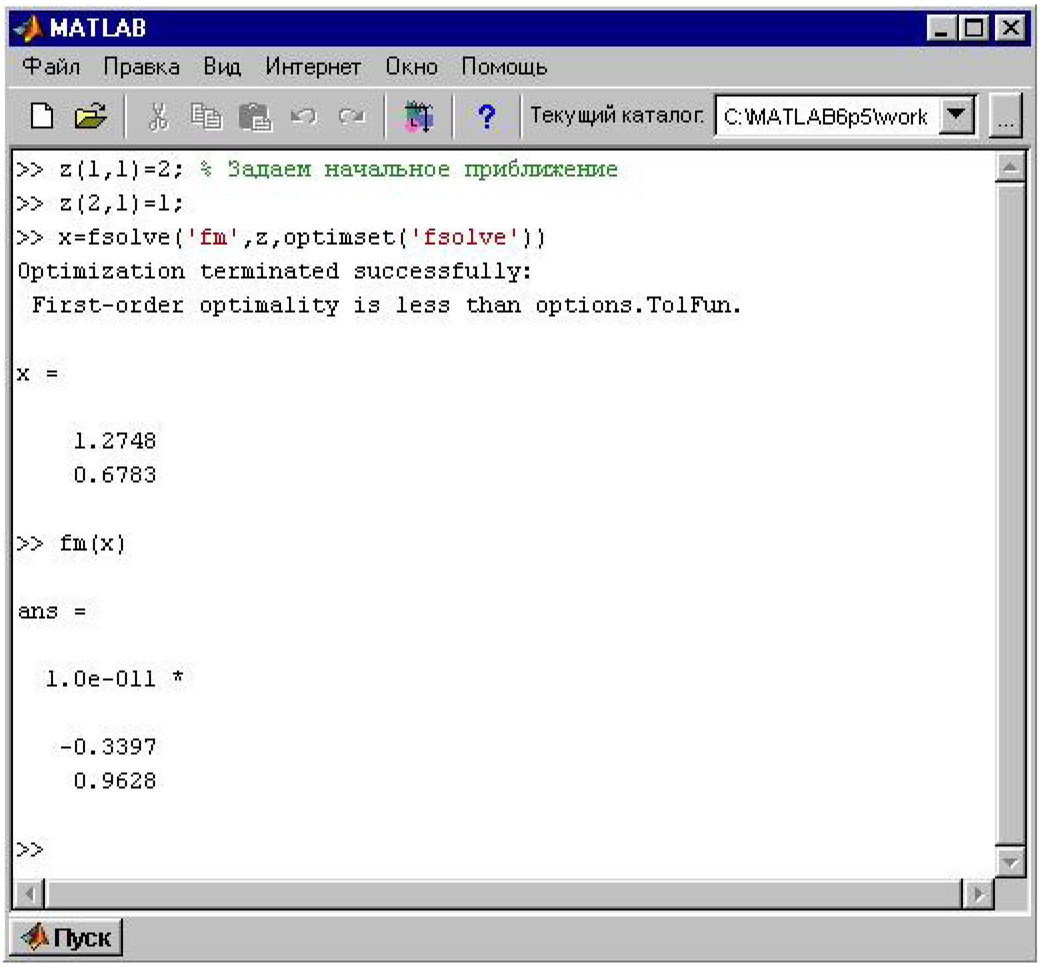
Для решения систем нелинейных уравнений средствами пакета Matlab применяется функция **fsolve(),** возвращающая вектор-столбец, компоненты которого являются корнями системы.

Пример 6: Решить систему уравнений (18) с помощью встроенной функции Matlab.

Файл Fm.m содержит описание системы уравнений.



Найдем решение системы и сделаем проверку.



Контрольные вопросы

1. Поясните геометрический смысл методов спуска.
2. Каким образом выбираются начальные приближения?
3. Почему предложенные методы являются итерационными?

4. Как выбор начального приближения влияет на сходимость метода Ньютона?

5. Что произойдет, если в окрестности решения нелинейной системы (15) функция (16) будет иметь несколько минимумов?

Индивидуальные задания

Найти решение системы уравнений с точностью $\varepsilon = 10^{-3}$ методом Ньютона. Начальное приближение найти графически. Определить число итераций и оценить погрешности решения.

Номер варианта	Система уравнений
1	$\begin{cases} 2x^3 - y^2 - 1 = 0, \\ x \cdot y^3 - y - 4 = 0. \end{cases}$
2	$\begin{cases} e^{x \cdot y} - x^2 + y - 1.06 = 0, \\ (x + 0.5)^2 + y^2 - 0.6 = 0, \end{cases} \quad x > 0,\ y > 0.$
3	$\begin{cases} y - 4 + \sqrt{1 - (x - 2.7)^2} = 0, \\ x + 2\sin(y - 5) - 1 = 0. \end{cases}$
4	$\begin{cases} (x - y)^3 - 8 \cdot (x + y) = 0, \\ 2 \cdot (x + y) + 15 \cdot \ln(x + y) - 5 = 0. \end{cases}$
5	$\begin{cases} x^2 + y^2 + z^2 - 1 = 0, \\ 2x^2 + y^2 - 4z = 0, \\ 3x^2 - 4y + z^2 - 1 = 0. \end{cases}$
6	$\begin{cases} tg(x \cdot y + 0.1) - x^2 = 0, \\ 0.6x^2 + 2y^2 - 1 = 0, \end{cases} \quad x > 0,\ y > 0.$
7	$\begin{cases} \cos(y + 1.5) - x - 0.4 = 0, \\ \sin(1.1 \cdot x) - 3y - 1 = 0. \end{cases}$
8	$\begin{cases} x^2 \cdot y^2 - 3x^3 - 6y^3 + 8 = 0, \\ x^4 - 9y + 2 = 0. \end{cases}$
9	$\begin{cases} 0.8x^2 + 2x \cdot y + 1.3y^2 + 20x - 15y = 0, \\ e^{0.6y - 0.8x} - 1.14x - 1.52y = 0. \end{cases}$
10	$\begin{cases} 2y - \cos(x + 0.9) - 0.2 = 0, \\ x + \sin(y - 0.5) + 0.6 = 0. \end{cases}$

§ 4 Интерполирование функций

Пусть функция $f(x)$ задана множеством своих значений для дискретного набора точек:

x_0	x_1	…	x_n
f_0	f_1	…	f_n

здесь $f_i = f(x_i)$.

Требуется найти интерполяционный многочлен $P(x) = P_n(x)$ степени не выше n, значения которого в узлах интерполяции x_i совпадают со значениями данной функции $P(x_i) = f_i$.

4.1 Интерполяционный полином Лагранжа

Для функции $f(x)$, заданной таблицей, построим интерполяционный многочлен $L_n(x)$, степень которого не выше n, в следующем виде:

$$L_n(x) = l_0(x) + l_1(x) + \ldots + l_n(x),$$

где $l_i(x)$ – многочлен степени n, причем

$$l_i(x_k) = \begin{cases} f_i, \ \text{если} \ i = k, \\ 0, \ \text{если} \ i \neq k. \end{cases} \tag{20}$$

Многочлены $l_i(x)$ составим следующим образом

$$l_i(x) = c_i(x - x_0)(x - x_1) \cdot \ldots \cdot (x - x_{i-1})(x - x_{i+1}) \cdot \ldots \cdot (x - x_n),$$

где c_i – постоянный коэффициент, значение которого находится из первой части условия (20)

$$c_i = \frac{f_i}{(x_i - x_0)(x_i - x_1) \cdot \ldots \cdot (x_i - x_{i-1})(x_i - x_{i+1}) \cdot \ldots \cdot (x_i - x_n)}.$$

Тогда интерполяционный многочлен Лагранжа имеет вид

$$L_n(x) = \sum_{i=1}^{n} f_i \frac{(x - x_0)(x - x_1) \cdot \ldots \cdot (x - x_{i-1})(x - x_{i+1}) \cdot \ldots \cdot (x - x_n)}{(x_i - x_0)(x_i - x_1) \cdot \ldots \cdot (x_i - x_{i-1})(x_i - x_{i+1}) \cdot \ldots \cdot (x_i - x_n)}.$$

4.2 Интерполяционные формулы Ньютона

Формулы Ньютона предназначены для таблиц с равноотстоящими узлами, т.е. $h = x_{i+1} - x_i$, $i = \overline{1,n}$. Определим разности между значениями функции в узлах интерполяции. Конечная разность первого порядка имеет вид

$$\Delta f_i = f_{i+1} - f_i .$$

Конечная разность второго порядка:

$$\Delta^2 f_i = \Delta f_{i+1} - \Delta f_i = f_{i+2} - 2 f_{i+1} + f_i .$$

Конечная разность n-го порядка вычисляется по формуле

$$\Delta^k f_i = f_{i+k} - k \cdot f_{i+k-1} + \frac{k(k-1)}{2!} f_{i+k-2} - \ldots + (-1)^k f_i .$$

Первая интерполяционная формула Ньютона используется для интерполирования и экстраполирования в точках x, близких к началу таблицы x_0, и имеет следующий вид:

$$P_n(x) = f_0 + t \cdot \Delta f_0 + \frac{t(t-1)}{2!} \Delta^2 f_0 + \ldots + \frac{t(t-1) \cdot \ldots \cdot (t-n+1)}{n!} \Delta^n f_0 ,$$

где $t = \dfrac{x - x_0}{h}$, число n выбирают так, чтобы конечные разности n-го порядка были практически постоянными.

Когда значение аргумент находится ближе к концу отрезка интерполяции x_n, используется вторая интерполяционная формула Ньютона

$$P_n(x) = f_n + q \cdot \Delta f_{n-1} + \frac{q(q+1)}{2!} \Delta^2 f_{n-2} + \ldots + \frac{q(q+1) \cdot \ldots \cdot (q+n-1)}{n!} \Delta^n f_0 ,$$

где $q = \dfrac{x - x_n}{h}$.

4.3 Сплайн-интерполяция

При большом количестве узлов интерполяции приходится использовать полиномы высокой степени. Можно этого избежать, разбив отрезок интерполяции на несколько частей и построив на каждой части свой интерполяционный многочлен. Существенный недостаток такого интерполирования состоит в том, что в точках сшивки разных интерполяционных полиномов их первая производная будет разрывной, поэтому для решения задачи кусочно-линейной интерполяции используют особый вид кусочно-полиномиальной интерполяции – сплайн-интерполяцию.

Сплайн – это функция, которая на каждом частичном отрезке интерполирования является алгебраическим многочленом, а на заданном отрезке непрерывна вместе с несколькими своими производными.

Пусть интерполируемая функция задана таблично. Длина частичного отрезка $[x_{i-1}, x_i]$

$$h_i = x_i - x_{i-1}, \ i = \overline{1, n}.$$

Найдем кубический сплайн на каждом из отрезков $[x_{i-1}, x_i]$:

$$S(x) = a_i + b_i(x - x_{i-1}) + c_i(x - x_{i-1})^2 + d_i(x - x_{i-1})^3, \tag{21}$$

где a_i, b_i, c_i, d_i – неизвестные коэффициенты.

Пусть значения $S(x)$ совпадают в узлах с табличными значениями функции $f(x)$

$$\begin{aligned} &S(x_{i-1}) = f_{i-1} = a_i, \\ &S(x_i) = f_i = a_i + b_i h_i + c_i h_i^2 + d_i h_i^3. \end{aligned} \tag{22}$$

Для получения дополнительных условий потребуем непрерывности первой и второй производных сплайна (21) во всех точках, включая узлы:

$$\begin{aligned} &\frac{dS(x)}{dx} = b_i + 2c_i(x - x_{i-1}) + 3d_i(x - x_{i-1})^2, \\ &\frac{d^2S(x)}{dx^2} = 2c_i + 6d_i(x - x_{i-1}). \end{aligned} \tag{23}$$

Найдем левые и правые производные (23) во внутреннем узле x_i:

$$\begin{aligned} &\frac{dS(x_i - 0)}{dx} = b_i + 2c_i h_i + 3d_i h_i^2, \quad \frac{dS(x_i + 0)}{dx} = b_{i+1}, \\ &\frac{d^2S(x_i - 0)}{dx^2} = 2c_i + 6d_i h_i, \qquad \frac{d^2S(x_i + 0)}{dx^2} = 2c_{i+1}, \ \ i = \overline{1, n-1}. \end{aligned} \tag{24}$$

Приравнивая найденные производные, получаем $2(n-1)$ условие. Потребуем нулевой кривизны сплайна на концах отрезка интерполирования

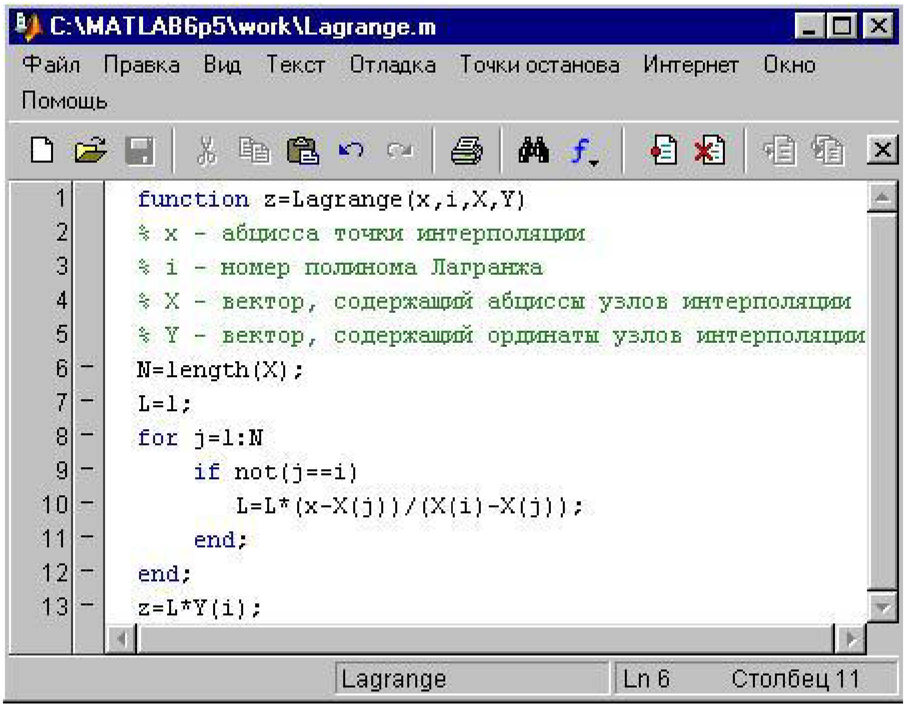
$$c_1 = 0, \ c_n + 3d_n h_n = 0. \tag{25}$$

Решая систему (22), (24), (25), найдем значения неизвестных коэффициентов, определяющих совокупность всех формул для искомого интерполяционного сплайна

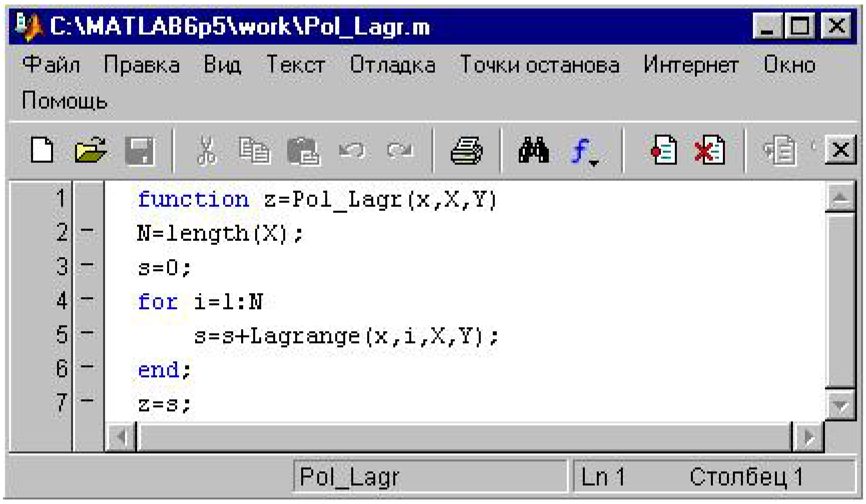
$$S_i(x) = f_{i-1} + b_i(x - x_{i-1}) + c_i(x - x_{i-1})^2 + d_i(x - x_{i-1})^3.$$

Пример 7: Решить задачу интерполяции с помощью многочлена Лагранжа для функции $f(x) = \sin(x)$, заданной таблично на интервале $[0, 2\pi]$ при $n = 8$.

1. Создадим функцию, возвращающую значение многочлена $l_i(x)$.

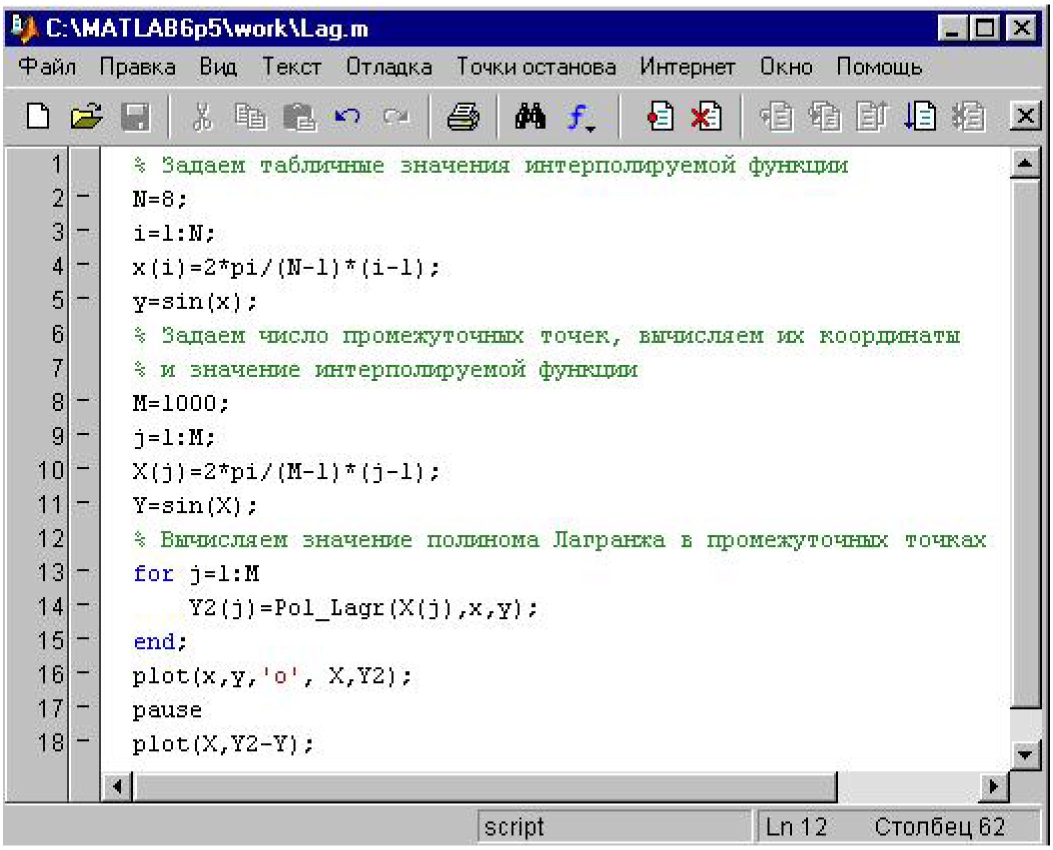


2. Создадим файл Pol_Lagr.m, возвращающий значение полинома Лагранжа.

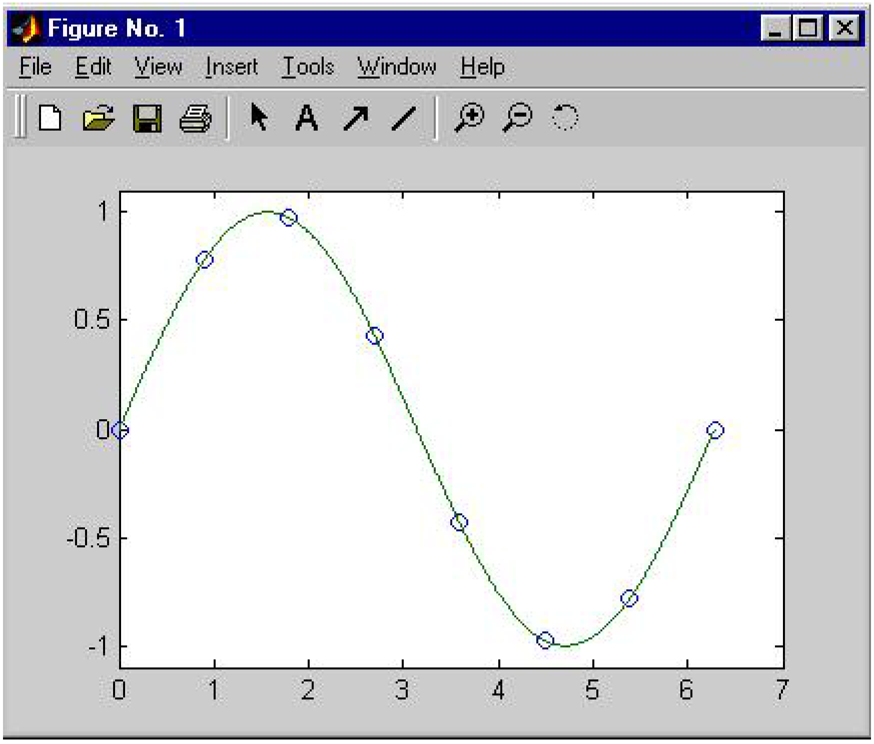


3. В файле задается табличная функция, вычисляются ее точные значения,

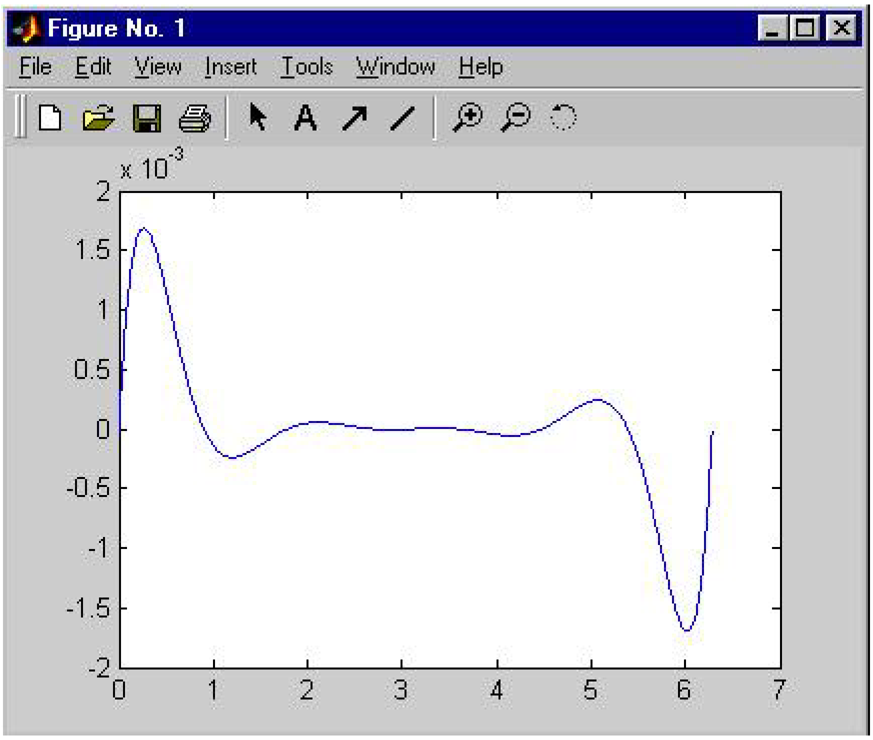
значения полинома Лагранжа.



Значения табличной функции и интерполированные значения функции представлены на графике.

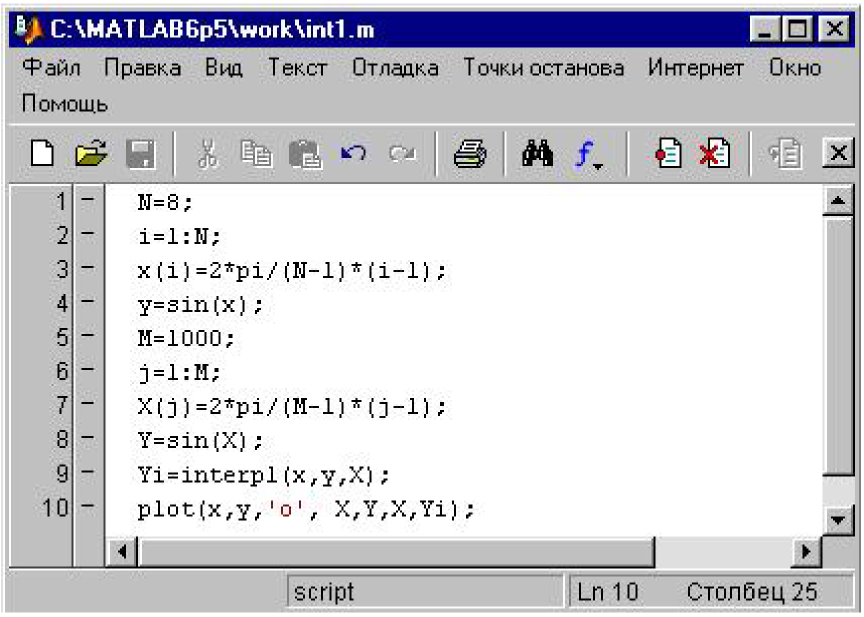


Погрешность аппроксимации функции $f(x)=\sin(x)$ полиномом Лагранжа.

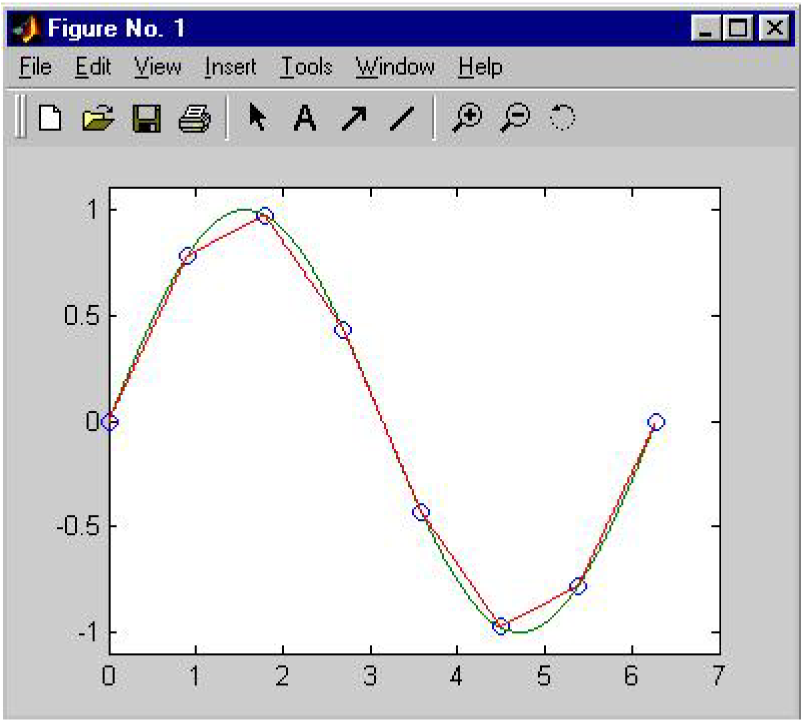


Пример 8: Решить задачу интерполяции для функции $f(x) = \sin(x)$, заданной таблично на интервале $[0,\ 2\pi]$ при $n = 8$ средствами пакета Matlab.

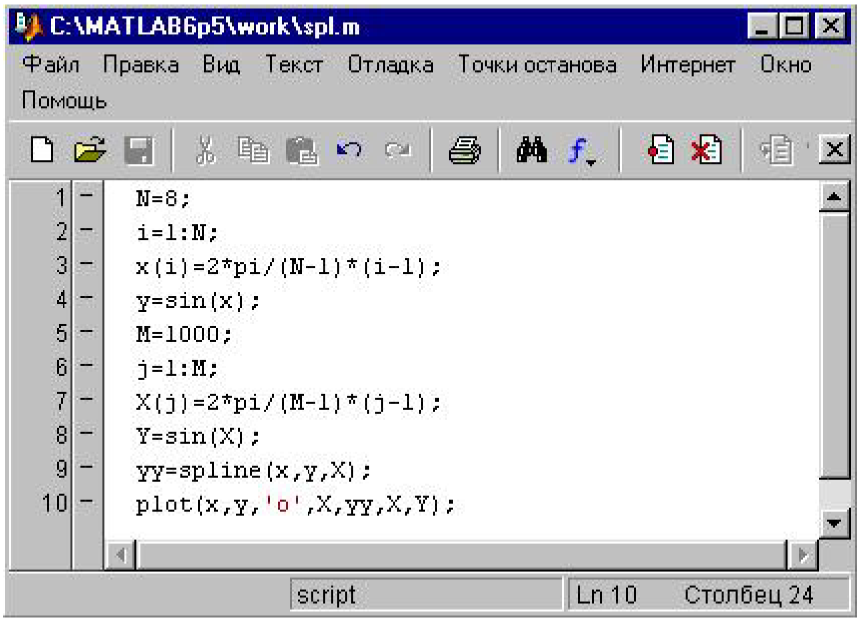
Для решения задачи одномерной интерполяции в пакете Matlab используется функция **interp1()**.



Исходные данные и результат линейной интерполяции представлены на рисунке.

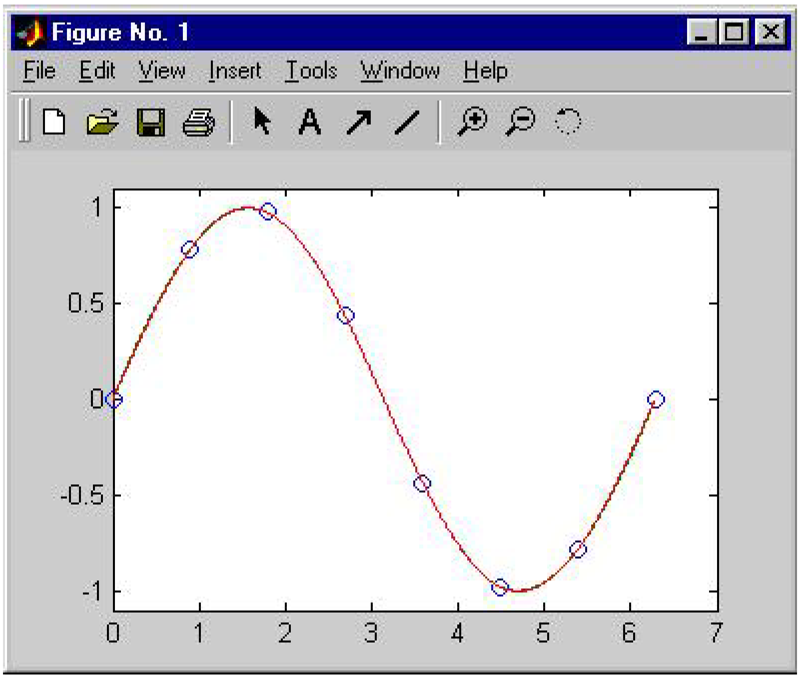


Решение задачи с помощью кубических сплайнов, используя функцию пакета Matlab **spline()**.




```
N=8;
i=1:N;
x(i)=2*pi/(N-1)*(i-1);
y=sin(x);
M=1000;
j=1:M;
X(j)=2*pi/(M-1)*(j-1);
Y=sin(X);
yy=spline(x,y,X);
plot(x,y,'o',X,yy,X,Y);
```

Визуализация исходных данных и результатов кубической сплайн-интерполяции представлена на графике.



Контрольные вопросы

1. Какие точки называются узлами интерполяции?

2. При решении каких задач используются интерполяционные формулы Ньютона?

3. Какие узлы называются равноотстоящими?

4. В каких случаях применяется сплайн-интерполяция?

5. В чем заключается основное отличие одномерной интерполяции от кубической?

Индивидуальные задания

1. Пользуясь первой и второй интерполяционными формулами Ньютона вычислить значение функции для указанных значений аргументов.

2. Решить задачу интерполяции средствами пакета Matlab с помощью линейных и кубических сплайнов. Привести графики исходных данных, результатов сплайн-интерполяции и погрешности аппроксимации.

Номер варианта	x	$f(x)$	Значения аргумента	Номер варианта	x	$f(x)$	Значения аргумента
1	2	3	4	5	6	7	8
1	1.50	0.51183	$x = 1.50911$ $x = 1.59513$	6	0.50	1.6487	$x = 0.50721$ $x = 0.56894$
	1.51	0.50624			0.51	1.6653	
	1.52	0.50064			0.52	1.6820	
	1.53	0.49503			0.53	1.6989	
	1.54	0.48940			0.54	1.7160	
	1.55	0.48376			0.55	1.7333	
	1.56	0.47811			0.56	1.7507	
	1.57	0.47245			0.57	1.7683	
	1.58	0.46678			0.58	1.7860	
	1.59	0.46110			0.59	1.8040	
	1.60	0.45540			0.60	1.8221	
2	0.00	0.28081	$x = 0.01928$ $x = 0.47113$	7	1.1	0.89121	$x = 1.1511$ $x = 2.0316$
	0.05	0.31270			1.2	0.93204	
	0.10	0.34549			1.3	0.96356	
	0.15	0.37904			1.4	0.98545	
	0.20	0.41318			1.5	0.99749	
	0.25	0.44774			1.6	0.99957	
	0.30	0.48255			1.7	0.99166	
	0.35	0.51745			1.8	0.97385	
	0.40	0.55226			1.9	0.94630	
	0.45	0.58682			2.0	0.90930	
	0.50	0.62096			2.1	0.86321	
3	1.0	0.5652	$x = 1.0113$ $x = 1.9592$	8	0.10	3.63004	$x = 0.1006$ $x = 2.5304$
	1.1	0.6375			0.35	3.75680	
	1.2	0.7147			0.60	3.88933	
	1.3	0.7973			0.85	4.03258	
	1.4	0.8861			1.10	4.19310	
	1.5	0.9817			1.35	4.38042	
	1.6	1.0848			1.60	4.60963	
	1.7	1.1964			0.85	4.90697	
	1.8	1.3172			2.10	5.32331	
	1.9	1.4482			2.35	5.97322	
	2.0	1.5906			2.60	7.18210	

1	2	3	4	5	6	7	8
4	0.50	1.6487	$x = 0.52301$ $x = 0.58967$	9	1.50	0.51183	$x = 1.50253$ $x = 1.59614$
	0.51	1.6653			1.51	0.50624	
	0.52	1.6820			1.52	0.50064	
	0.53	1.6989			1.53	0.49503	
	0.54	1.7160			1.54	0.48940	
	0.55	1.7333			1.55	0.48376	
	0.56	1.7507			1.56	0.47811	
	0.57	1.7683			1.57	0.47245	
	0.58	1.7860			1.58	0.46678	
	0.59	1.8040			1.59	0.46110	
	0.60	1.8221			1.60	0.45540	
5	0.10	3.63004	$x = 0.10056$ $x = 2.57321$	10	0.00	0.28081	$x = 0.02475$ $x = 0.48675$
	0.35	3.75680			0.05	0.31270	
	0.60	3.88933			0.10	0.34549	
	0.85	4.03258			0.15	0.37904	
	1.10	4.19310			0.20	0.41318	
	1.35	4.38042			0.25	0.44774	
	1.60	4.60963			0.30	0.48255	
	0.85	4.90697			0.35	0.51745	
	2.10	5.32331			0.40	0.55226	
	2.35	5.97322			0.45	0.58682	
	2.60	7.18210			0.50	0.62096	

§ 5 Обработка экспериментальных данных

Пусть в результате измерений получена таблица некоторой зависимости $f(x)$:

x	x_1	x_2	…	x_n
$F(x)$	y_1	y_2	…	y_n

Требуется найти формулу, выражающую данную зависимость аналитически. Один из подходов состоит в построении интерполяционного многочлена, значения которого в узлах интерполяции x_i совпадают со значениями данной функции $f(x_i)=f_i$.

Если значения функции $f(x)$ известны с некоторой погрешностью, то требование совпадение значений не оправдано, поскольку оно не означает совпадение характеров исходной и интерполирующей функции. Поэтому поставим задача так: найти функцию вида

$$y=F(x),$$

которая в точках $x_1, x_2, \ldots, x_n$ принимает значения, близкие к табличным значениям $y_1, y_2, \ldots, y_n$.

Предположим, что приближающая функция $F(x)$ в точках $x_1, x_2, \ldots, x_n$ имеет значения $\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n$. Тогда нужно найти функцию $F(x)$ определенного вида так, чтобы сумма квадратов

$$(y_1-\tilde{y}_1)^2+(y_2-\tilde{y}_2)^2+\ldots+(y_n-\tilde{y}_n)^2$$

была наименьшей.

5.1 Нахождение приближающей функции в виде линейной

Приближающая функция имеет вид

$$F(x)=ax+b, \tag{26}$$

где a, b – параметры.

Находим частные производные

$$\frac{\partial F}{\partial a}=x,\ \ \frac{\partial F}{\partial b}=1.$$

Составляем систему уравнений

$$\sum_{i=1}^{n}\left(y_i-ax_i-b\right)x_i=0,$$

$$\sum_{i=1}^{n}\left(y_i-ax_i-b\right)=0.$$

Разделив каждое уравнение на n, получаем

$$\left(\frac{1}{n}\cdot\sum_{i=1}^{n}x_i^2\right)\cdot a+\left(\frac{1}{n}\cdot\sum_{i=1}^{n}x_i\right)\cdot b=\frac{1}{n}\cdot\sum_{i=1}^{n}x_i y_i,$$

$$\left(\frac{1}{n}\cdot\sum_{i=1}^{n}x_i\right)\cdot a+b=\frac{1}{n}\cdot\sum_{i=1}^{n}y_i.$$

Вычислив значения параметров a, b, получаем конкретный вид функции (26).

Существует возможность с помощью подходящего преобразования переменных получить линейную зависимость для следующих функций.

Функция $y=f(x)$	Линеаризованная форма, $Y=ax+b$	Замена переменных и постоянных
$y=\frac{a}{x}+b$	$y=a\cdot\frac{1}{x}+b$	$X=\frac{1}{x},\ Y=y$
$y=\frac{d}{x+c}$	$y=-\frac{1}{c}\cdot xy+\frac{d}{c}$	$X=xy,\ Y=y,$ $c=-\frac{1}{a},\quad d=-\frac{b}{a}$
$y=\frac{1}{ax+b}$	$\frac{1}{y}=ax+b$	$X=x,\ Y=\frac{1}{y}$
$y=\frac{x}{ax+b}$	$\frac{1}{y}=a\cdot\frac{1}{x}+b$	$X=\frac{1}{x},\ Y=\frac{1}{y}$
$y=a\cdot\ln(x)+b$	$y=a\cdot\ln(x)+b$	$X=\ln(x),\ Y=y$
$y=c\cdot e^{ax}$	$\ln(y)=ax+\ln(c)$	$X=x,\ Y=\ln(y),\ c=e^b$
$y=c\cdot x^a$	$\ln(y)=a\cdot\ln(x)+\ln(c)$	$X=\ln(x),\ Y=\ln(y),$ $c=e^b$

$y=\dfrac{1}{(ax+b)^2}$	$\dfrac{1}{\sqrt{y}}=ax+b$	$X=x,\ Y=\dfrac{1}{\sqrt{y}}$
$y=\dfrac{c\cdot x}{e^{dx}}$	$\ln\left(\dfrac{y}{x}\right)=-dx+\ln(c)$	$X=x,\ Y=\ln\left(\dfrac{y}{x}\right),$ $c=e^b,\ d=-a$
$y=\dfrac{1}{1+c\cdot e^{ax}}$	$\ln\left(\dfrac{1}{y}-1\right)=ax+\ln(c)$	$X=x,\ Y=\ln\left(\dfrac{1}{y}-1\right),$ $c=e^b$

5.2 Нахождение приближающей функции в виде квадратного трехчлена

Приближающая функция имеет вид

$$F(x)=ax^2+bx+c, \tag{27}$$

где a, b, c – параметры.

Находим частные производные

$$\frac{\partial F}{\partial a}=x^2,\ \frac{\partial F}{\partial b}=x,\ \frac{\partial F}{\partial c}=1.$$

Составляем систему уравнений

$$\sum_{i=1}^{n}\left(y_i-ax_i^2-bx_i-c\right)x_i^2=0,$$

$$\sum_{i=1}^{n}\left(y_i-ax_i^2-bx_i-c\right)x_i=0,$$

$$\sum_{i=1}^{n}\left(y_i-ax_i^2-bx_i-c\right)=0.$$

Разделив каждое уравнение на n, имеем

$$\left(\frac{1}{n}\cdot\sum_{i=1}^{n}x_i^4\right)\cdot a+\left(\frac{1}{n}\cdot\sum_{i=1}^{n}x_i^3\right)\cdot b+\left(\frac{1}{n}\cdot\sum_{i=1}^{n}x_i^2\right)\cdot c=\frac{1}{n}\cdot\sum_{i=1}^{n}x_i^2y_i,$$

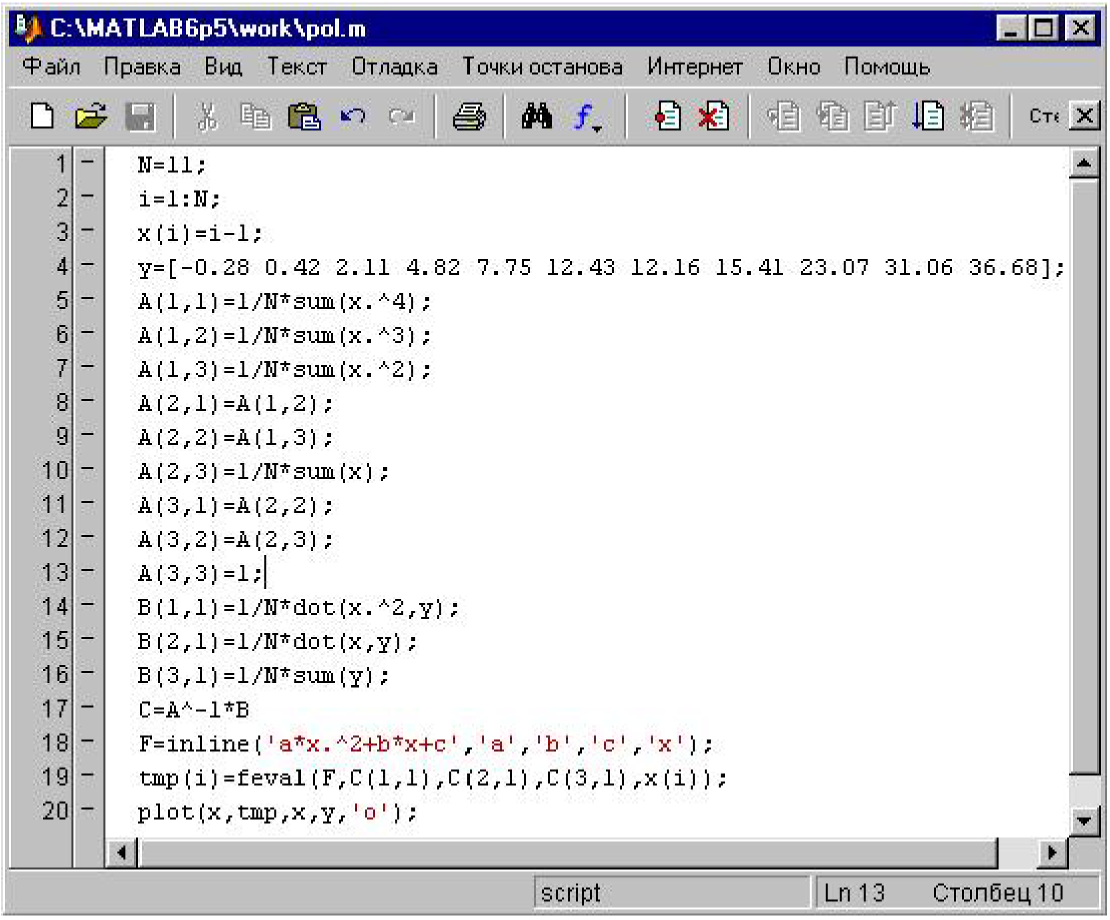
$$\left(\frac{1}{n}\cdot\sum_{i=1}^{n}x_i^3\right)\cdot a+\left(\frac{1}{n}\cdot\sum_{i=1}^{n}x_i^2\right)\cdot b+\left(\frac{1}{n}\cdot\sum_{i=1}^{n}x_i\right)\cdot c=\frac{1}{n}\cdot\sum_{i=1}^{n}x_iy_i$$

$$\left(\frac{1}{n}\cdot\sum_{i=1}^{n}x_i^2\right)\cdot a+\left(\frac{1}{n}\cdot\sum_{i=1}^{n}x_i\right)\cdot b+c=\frac{1}{n}\cdot\sum_{i=1}^{n}y_i.$$

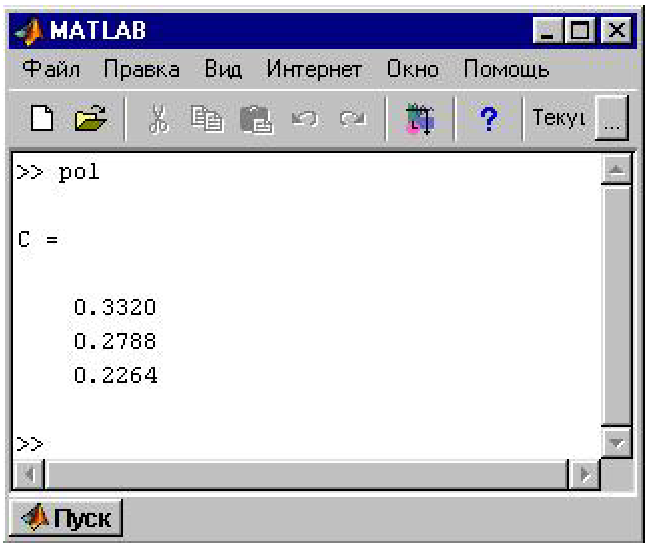
Решив систему относительно неизвестных a, b, c, находим значения параметров приближающей функции (27).

Пример 9: Даны табличные значения квадратичной зависимости. Найти коэффициенты квадратичной аппроксимирующей функции (27).

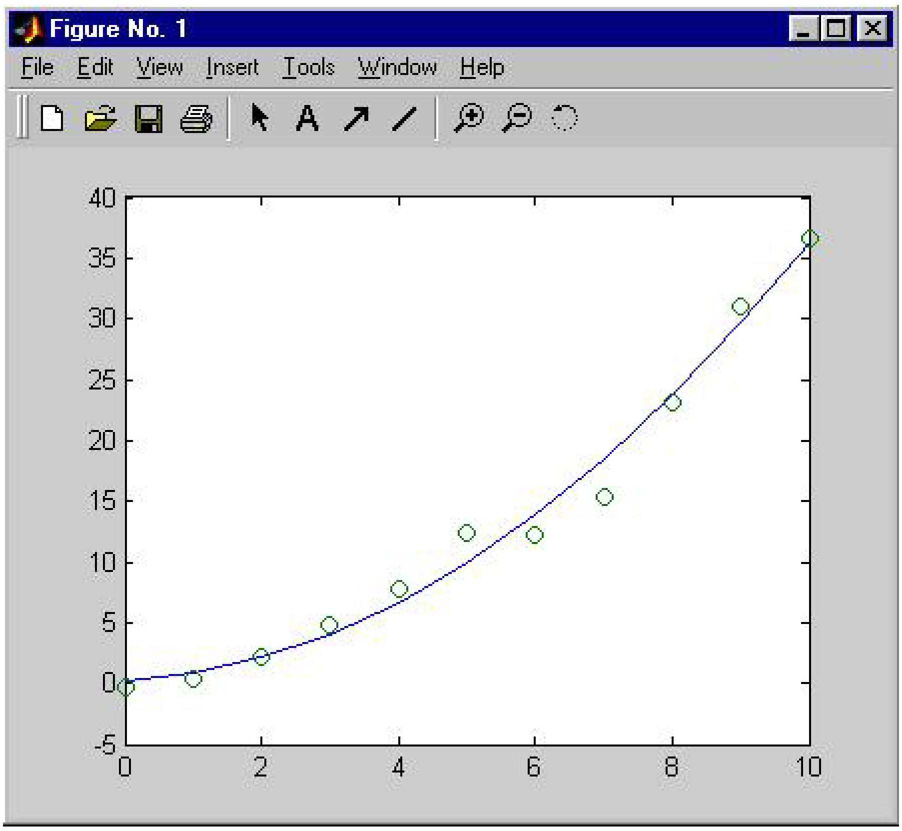
x	0	1	2	3	4	5	6	7	8	9	10
y	- 0.28	0.42	2.11	4.82	7.75	12.43	12.16	15.41	23.07	31.06	36.68



В результате работы программы получаем значение параметров приближающей функции (27).



На графике представлены исходные данные и аппроксимирующая функция.



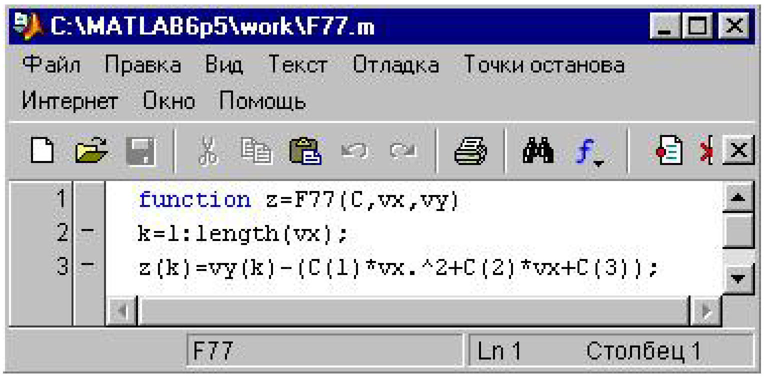
Пример 10: Используя табличные значения примера 9, найти параметры квадратичной аппроксимирующей функции (27) с помощью встроенной функции Matlab.

Для решения задачи обобщенной нелинейной регрессии в пакете Matlab имеется функция **lsqnonlin()**, возвращающая решение задачи нахождения точки минимума функции:

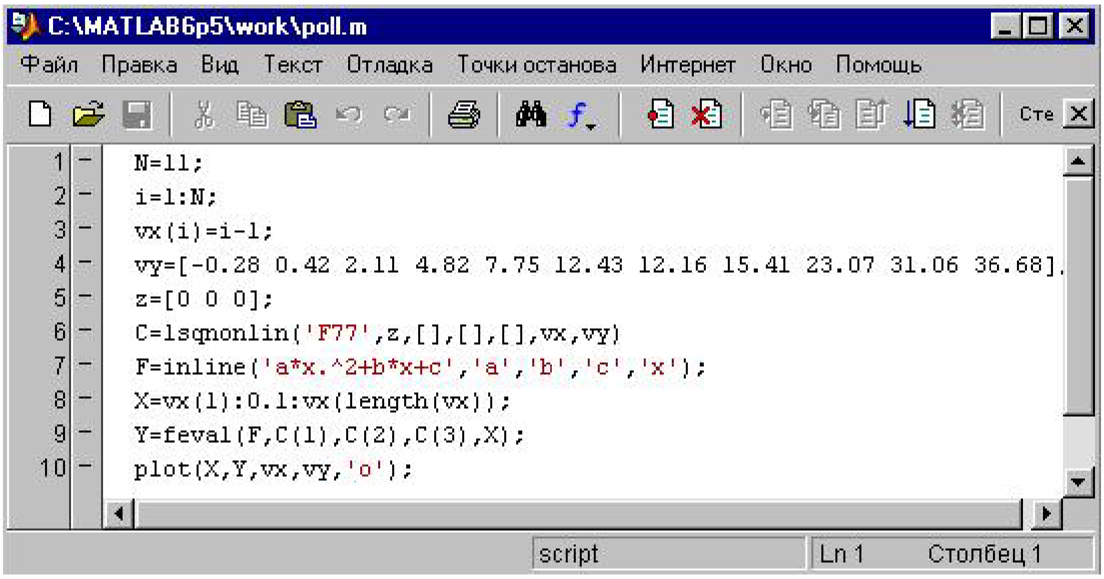
$$\min_x (f(x)) = f_1^2(x) + f_2^2(x) + \ldots + f_n^2(x) + L,$$

где $f(x)$ – вектор-функция, x – столбец искомых переменных, L – искомая константа.

1. Создадим файл F77.m, содержащий функцию, которая возвращает значение вектор-функции $f(x)$.



2. Вычислим значения аппроксимирующей функции с помощью функции lsqnonlin() и визуализируем исходные данные и аппроксимирующую функцию.



Параметры приближающей функции:

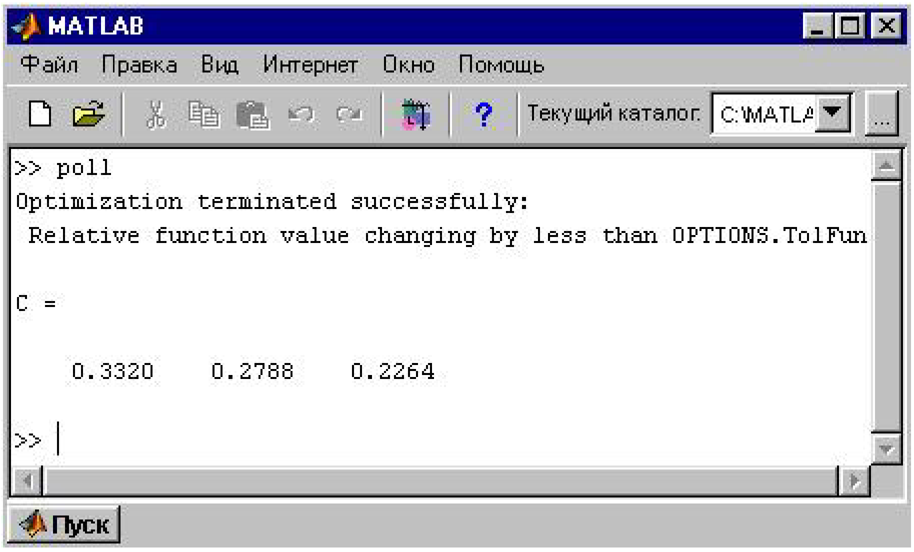
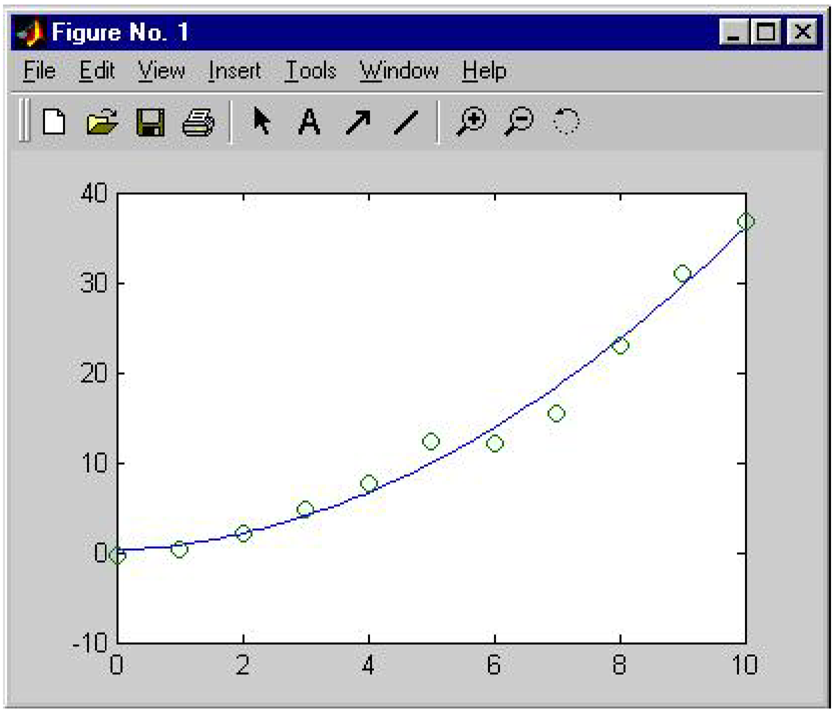


График исходных данных и приближающей функции:



Контрольные вопросы

1. Как можно добиться повышения качества приближения?
2. Какая ошибка является среднеквадратической?
3. За счет чего возникает полиномиальное раскачивание?

Индивидуальные задания

1. Осуществите замену переменных и получите линеаризованную форму для каждой из приведенных функций. С помощью метода наименьших квадра-

тов найдите значения параметров приближающей линейной функции. Постройте графики исходных данных и полученной аппроксимирующей функции.

2. Найдите аппроксимирующую функцию средствами пакета Matlab. Сравните с результатом, полученным методом наименьших квадратов. Рассчитайте значение среднеквадратического отклонения.

Номер варианта	Функция $y = f(x)$	Исходные данные		Номер варианта	Функция $y = f(x)$	Исходные данные	
		x	y			x	y
1	$y = c \cdot e^{ax}$	0	1.5	6	$y = a \cdot \ln(x) + b$	1.5	-0.72
		1	2.5			2.5	0.33
		2	3.5			3.5	1.09
		3	5			4	1.28
		4	7.5			5	1.85
2	$y = c \cdot x^{a}$	1	0.6	7	$y = \dfrac{x}{ax+b}$	1	0.03
		2	1.9			2	0.54
		3	4.3			3	0.28
		4	7.6			4	0.55
		5	12.6			5	0.78
3	$y = \dfrac{1}{ax+b}$	-1	6.62	8	$y = \dfrac{c \cdot x}{e^{dx}}$	1.6	3.36
		0	3.94			1.8	3.35
		1	2.17			2.1	3.19
		2	1.35			2.3	2.79
		3	0.89			2.9	2.67
4	$y = \dfrac{1}{(ax+b)^2}$	-1	13.45	9	$y = \dfrac{d}{x+c}$	2.4	1.30
		0	3.01			2.9	1.14
		1	0.67			3.1	1.24
		2	0.15			3.6	1.16
5	$y = \dfrac{a}{x} + b$	1	5.51	10	$y = \dfrac{1}{1 + c \cdot e^{ax}}$	0.1	0.90
		1.5	4.72			0.4	0.71
		1.9	4.54			0.7	0.95
		2.2	4.33			1.3	0.92
		2.5	4.02			1.9	1.14

§ 6 Численное дифференцирование и интегрирование

6.1 Численное дифференцирование аналитически заданных функций

По определению производная функции $f(x)$ равна:

$$f'(x) = \lim_{\Delta x \to 0} \frac{f(x+\Delta x) - f(x)}{\Delta x}.$$

Переходя от бесконечно малых разностей к конечным, получаем приближенную формулу численного дифференцирования:

$$f'(x) \approx \frac{f(x+\Delta x) - f(x)}{\Delta x}. \qquad (28)$$

Используя разложение функции $f(x)$ в ряд Тейлора, можно записать

$$f'(x) \approx f'(x) + \frac{f''(x)}{2!}\Delta x + \ldots \qquad (29)$$

Согласно формуле (29), основной член погрешности равен $\frac{f''(x)}{2!}\Delta x$, т.е. формула (28) имеет первый порядок точности по Δx.

Используя симметричную разностную схему можно записать следующее:

$$f'(x) \approx \frac{f(x+\Delta x) - f(x-\Delta x)}{2 \cdot \Delta x}. \qquad (30)$$

Используя разложение в ряд Тейлора, получаем

$$f'(x) \approx f'(x) + \frac{f'''(x)}{3!}(\Delta x)^2 + \ldots \qquad (31)$$

Формула (31) имеет второй порядок точности по Δx.

6.2 Численное дифференцирование функций, заданных таблицей

Пусть функция $f(x)$ задана таблично в конечном числе точек отрезка $[a,b]$. Требуется определить значение производной в некоторой точке отрезка $[a,b]$.

Выбрав $(n+1)$ узлов, заменим функцию $f(x)$ интерполяционным много членом $P_n(x)$. Тогда производная от этого многочлена применяется для приближенного представления производной функции $f(x)$:

$f'(x) \approx P_n'(x).$

Как правило, формулы численного дифференцирования применяют для нахождения производных в узлах x_i. Дифференцирование интерполяционных многочленов Ньютона в точке x_0 приводит к формуле:

$$f'(x_0) \approx \frac{1}{h}\left(\Delta f_0 - \frac{1}{2}\Delta^2 f_0 + \frac{1}{3}\Delta^3 f_0 - \ldots + \frac{(-1)^{n-1}}{n}\Delta^n f_0\right), \tag{32}$$

$$f'(x_0) \approx \frac{1}{h}\left(\Delta f_{-1} + \frac{1}{2}\Delta^2 f_{-2} + \frac{1}{3}\Delta^3 f_{-3} - \ldots + \frac{1}{n}\Delta^n f_{-n}\right). \tag{33}$$

Формула (32) применяется для начальных строк таблицы, формула (33) – для последних строк.

Для середины таблицы применяют формулы центрированной разности второго порядка точности по h.

$$f'(x_0) \approx \frac{f_1 - f_{-1}}{2h},$$

$$f''(x) \approx \frac{f_1 - 2f_0 + f_{-1}}{h^2},$$

$$f'''(x) \approx \frac{f_2 - f_1 + 2f_{-1} - f_{-2}}{2h^3},$$

где $f_k = f(x_0 + kh)$, $k = -2, -1, 0, 1, 2.$

6.3 Численное интегрирование методом прямоугольников

Геометрический смысл определенного интеграла

$$F = \int_a^b f(x)dx$$

– площадь фигуры, ограниченной графиком функции $f(x)$ и прямыми $x = a$, $x = b$.

Раздели отрезок $[a,b]$ на n равных отрезков длиной Δx

$$\Delta x = \frac{b-a}{n}.$$

Координата правого конца i-го отрезка определяется по формуле

$$x_i = x_0 + i \cdot \Delta x,$$

где $x_0 = a$, $i = \overline{0,n}$.

Простейшая оценка площади кривой может быть получена как сумма площадей прямоугольников, одна из сторон которого равна длине отрезка $[x_i, x_{i+1}]$, а высота равна значению $f(x_i)$ – метод левых прямоугольников (см. рисунок 9), или $f(x_{i+1})$ – метод правых прямоугольников (см. рисунок 10).

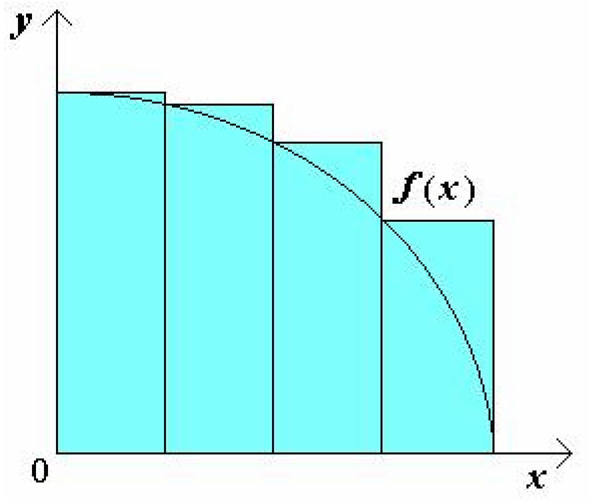


Рисунок 9 – Геометрическая интерпретация метода левых прямоугольников.

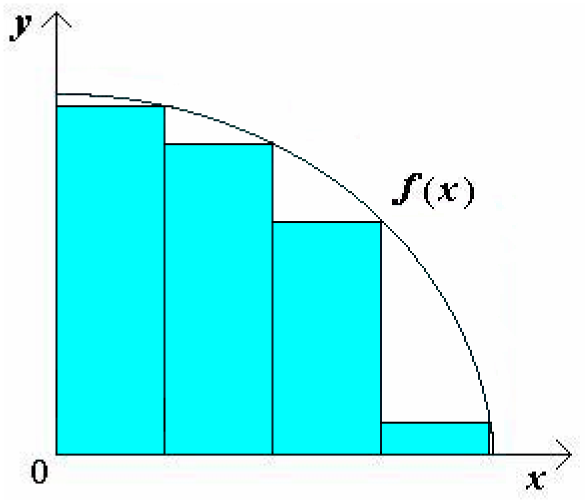


Рисунок 10 – Геометрическая интерпретация метода правых прямоугольников.

Тогда для левых прямоугольников определенный интеграл вычисляется по формуле

$$F_L = \sum_{i=0}^{n-1} f(x_i) \cdot \Delta x, \tag{34}$$

а для правых прямоугольников – по формуле

$$F_R = \sum_{i=1}^{n} f(x_i) \cdot \Delta x. \quad (35)$$

Погрешность метода левых и правых прямоугольников пропорциональна n^{-1}.

6.4 Численное интегрирование методом трапеций

Заменим функцию на каждом интервале $[x_i, x_{i+1}]$, $i = \overline{0, n-1}$ отрезком прямой, проходящей через точки $(x_i, f(x_i))$ и $(x_{i+1}, f(x_{i+1}))$. Тогда фигура, ограниченная графиком функции и прямыми $x = x_i$, $x = x_{i+1}$, является трапецией.

Тогда определенный интеграл определяется как сумма площадей всех трапеций по формуле:

$$F_n = \frac{1}{2}\sum_{i=0}^{n-1}\left(f(x_{i+1}) + f(x_i)\right) \cdot \Delta x = \left[\frac{1}{2} f(a) + \sum_{i=1}^{n-1} f(x_i) + \frac{1}{2} f(b)\right] \cdot \Delta x. \quad (36)$$

Полная погрешность формулы трапеций на отрезке $[a, b]$ по порядку величины равна $\mathrm{O}(n^{-2})$.

6.5 Численное интегрирование методом Симпсона

Формула получается при использовании параболической интерполяции по трем соседним точкам

$$y = ax^2 + bx + c. \quad (37)$$

Для нахождения параметров a, b, c полинома, проходящего через три точки (x_0, y_0), (x_1, y_1), (x_2, y_2) решаем систему линейных уравнений

$$\begin{cases} y_0 = ax_0^2 + bx_0 + c, \\ y_1 = ax_1^2 + bx_1 + c, \\ y_2 = ax_2^2 + bx_2 + c. \end{cases} \quad (38)$$

Решив систему (38), подставим найденные значения в (37) и получим

$$y = y_0 \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} + y_1 \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} + y_0 \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}. \quad (39)$$

Интегрируя (39) на отрезке $[x_0, x_2]$, находим

$$F_0 = \frac{1}{3}\left(y_0 + 4y_1 + y_2\right)\cdot \Delta x,$$

где $\Delta x = x_1 - x_0 = x_2 - x_1$.

Искомый определенный интеграл находится как площадь всех параболических сегментов:

$$F_n = \frac{1}{3}\left[f(a) + 4f(x_1) + 2f(x_2) + 4f(x_3) + \ldots + 2f(x_{n-2}) + 4f(x_{n-1}) + f(b)\right]\cdot \Delta x.$$

В формуле Симпсона число n должно быть четным.

Полная погрешность формулы Симпсона на отрезке $[a,b]$ по порядку величины составляет $\mathrm{O}\left(n^{-4}\right)$.

6.6 Численное интегрирование методом трех восьмых

$$\begin{aligned} F_n = \frac{3\cdot\Delta x}{8}&\left[f(a) + f(x_{3m}) + 2\left(f(x_3) + f(x_6) + \ldots + f(x_{3m-3})\right) + \right. \\ &\left. + 3\left(f(x_1) + f(x_2) + f(x_4) + f(x_5) + \ldots + f(x_{3m-2}) + f(x_{3m-1})\right)\right], \end{aligned} \tag{40}$$

где $\Delta x = \dfrac{b-a}{n} = \dfrac{b-a}{3m}$.

В формуле (40) число n должно быть равно $3m$.

Значение полной погрешности правила трех восьмых по порядку величины совпадает со значением полной погрешности формулы Симпсона.

Заметим, что оценка погрешности вычисления интеграла от функции, зависящей от d переменных, определяется следующим образом: если для одномерного случая погрешность составляет $\mathrm{O}\left(n^{-\alpha}\right)$, то в d-мерном случае она равна $\mathrm{O}\left(n^{-\frac{\alpha}{d}}\right)$.

6.7 Численное интегрирование методом Монте-Карло

Пусть существует прямоугольник высотой H и длиной $(b-a)$ такой, что функция $f(x)$ целиком лежит внутри прямоугольника. Сгенерируем n пар случайных чисел, равномерно распределенных в данном прямоугольнике:

$$a \le x_i \le b, \quad 0 \le y_i \le H.$$

Доля точек (x_i, y_i), удовлетворяющих условию $y_i \le f(x_i)$, является оценкой отношения интеграла от функции $f(x)$ к площади рассматриваемого прямоугольника. Оценка интеграла может быть получена по формуле

$$F_n = A\frac{n_s}{n}, \tag{41}$$

где n_s – число точек, удовлетворяющих условию $y_i \le f(x_i)$, A – площадь прямоугольника.

Можно вычислить определенный интеграл как среднее значение функции $f(x)$ на отрезке $[a,b]$:

$$F_n = \frac{(b-a)}{n}\sum_{i=1}^{n} f(x_i), \tag{42}$$

где x_i – последовательность случайных чисел с равномерным законом распределения на отрезке $[a,b]$.

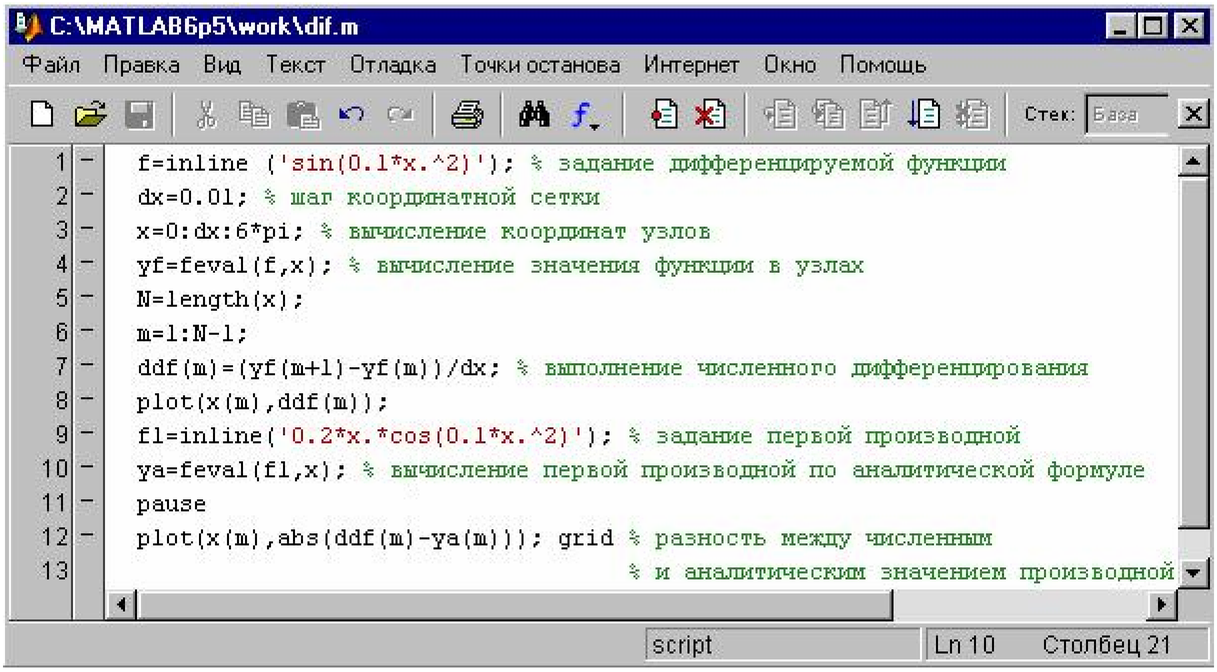
В отличие от рассмотренных выше методов, погрешность метода Монте-Карло не зависит от размерности подынтегральной функции d и составляет $\mathrm{O}(n^{-0.5})$. Поэтому при достаточно больших d интегрирование по методу Монте-Карло приводит к меньшим погрешностям, при тех же значениях n.

Пример 11: Вычислить значение производной функции

$$f(x) = \sin(0.1 \cdot x^2) \tag{43}$$

на отрезке $[0, 6\pi]$.

Листинг программы, численно дифференцирующая аналитически заданную функцию (43), представлен в следующем окне.



```
f=inline ('sin(0.1*x.^2)'); % задание дифференцируемой функции
dx=0.01; % шаг координатной сетки
x=0:dx:6*pi; % вычисление координат узлов
yf=feval(f,x); % вычисление значения функции в узлах
N=length(x);
m=1:N-1;
ddf(m)=(yf(m+1)-yf(m))/dx; % выполнение численного дифференцирования
plot(x(m),ddf(m));
f1=inline('0.2*x.*cos(0.1*x.^2)'); % задание первой производной
ya=feval(f1,x); % вычисление первой производной по аналитической формуле
pause
plot(x(m),abs(ddf(m)-ya(m))); grid % разность между численным
                                   % и аналитическим значением производной
```

Визуализация производной функции (43) и разность между численными и аналитическими значениями первой производной с шагом сетки $dx = 0.01$ изображены на рисунках 11, 12, а разность между численными и аналитическими значениями производной с шагом сетки $dx = 0.0001$ на рисунке 13.

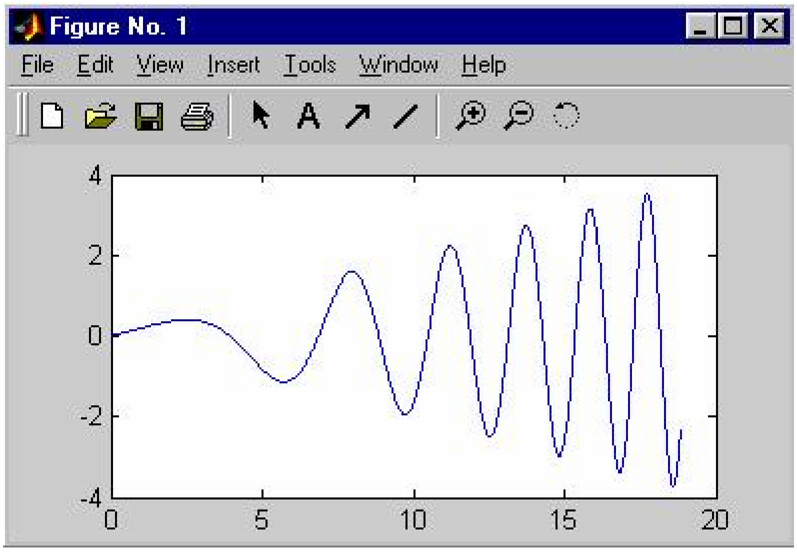


Рисунок 11

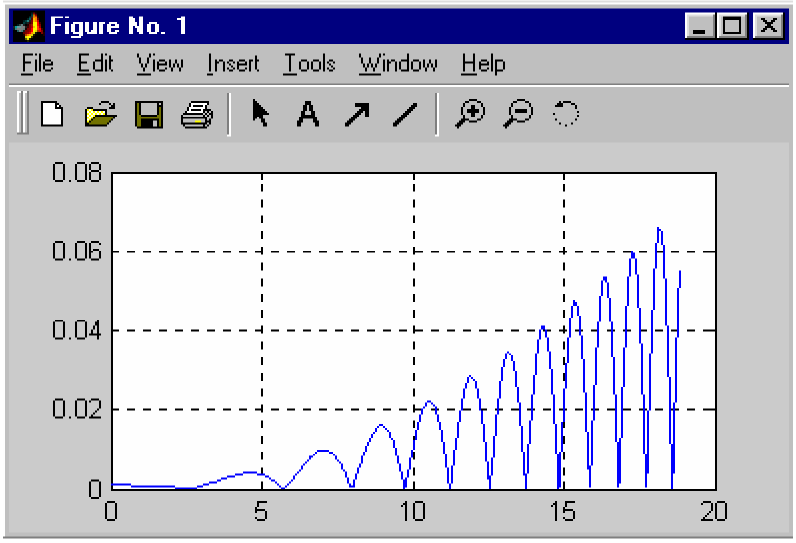


Рисунок 12.

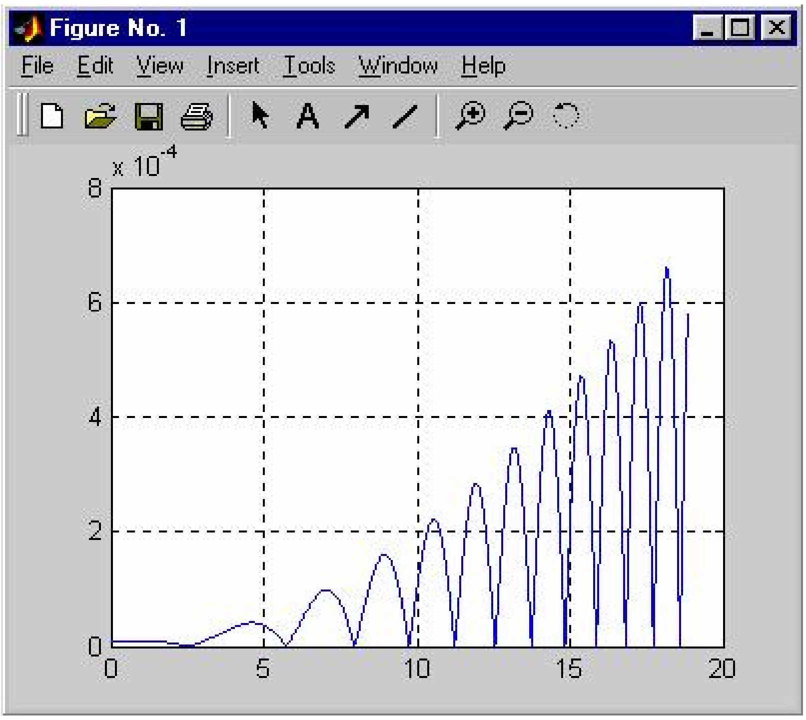


Рисунок 13.

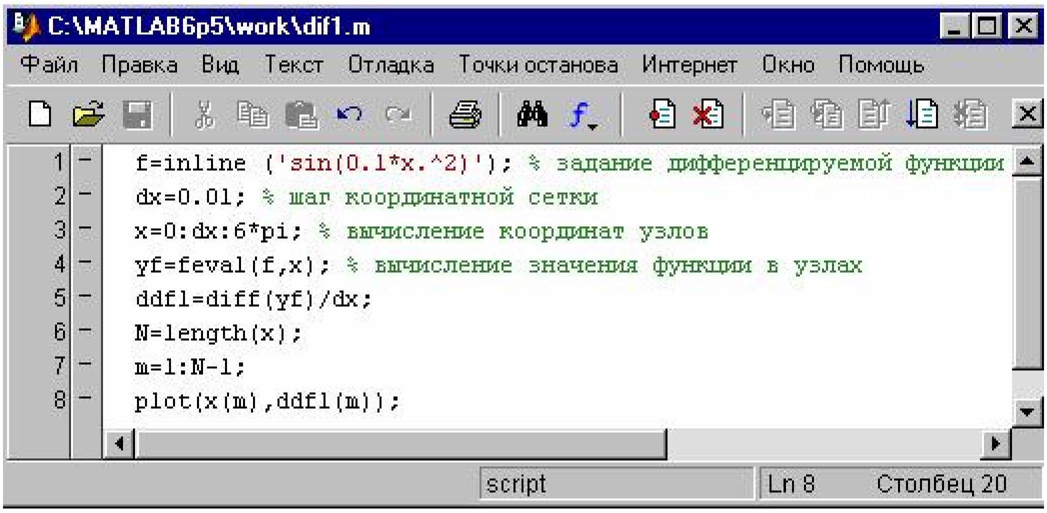
Для аппроксимации производных конечными разностями в пакете Matlab существует функция **diff()**:

diff(x) – возвращает конечные разности, вычисленные по смежным элементам вектора x;

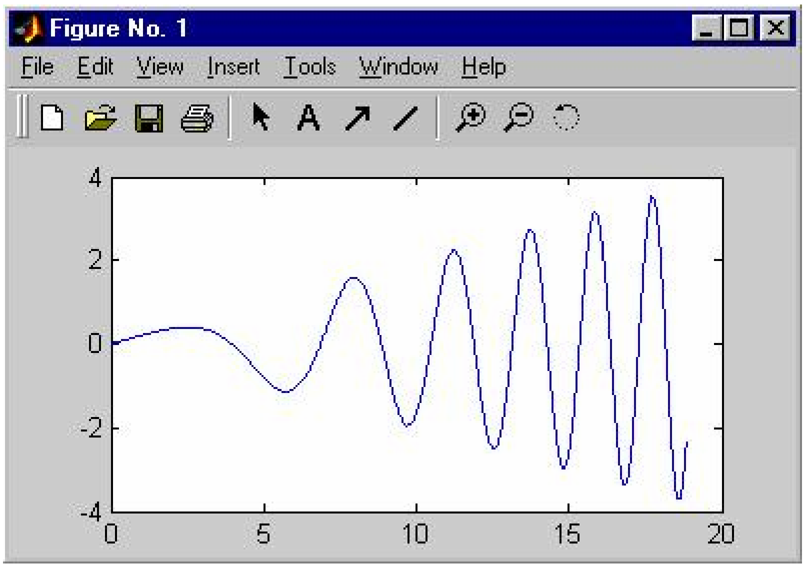
diff(x, n) – возвращает конечные разности n-го порядка, вычисленные по смежным элементам вектора x.

Пример 12: Вычислить значение производной функции (43) на отрезке $[0, 6\pi]$ с использованием соответствующей функции Matlab.

Программа, вычисляющая значение производной с помощью конечных разностей представлена в окне.



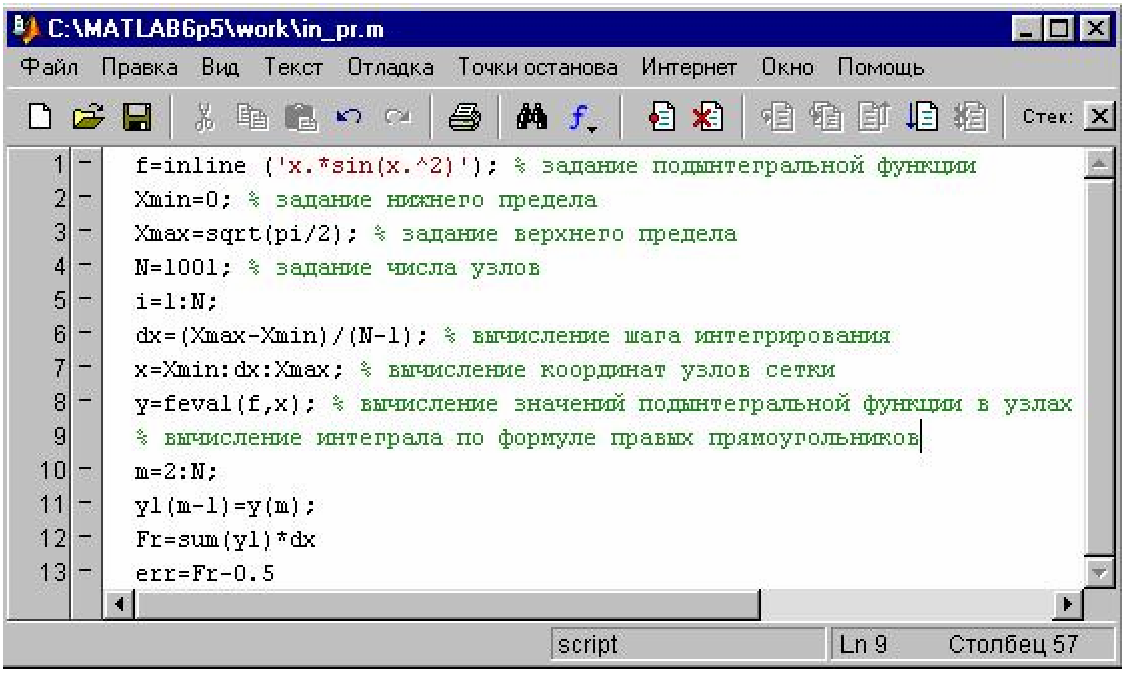
Визуализация первой производной функции (43), полученной с помощью функции diff().



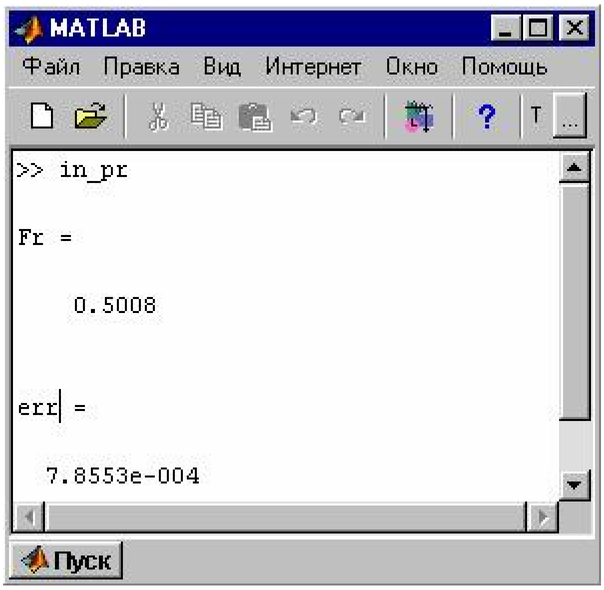
Пример 13: Вычислить значение определенного интеграла

$$\int_0^{\sqrt{\frac{\pi}{2}}} x \cdot \sin\left(x^2\right) dx \qquad (44)$$

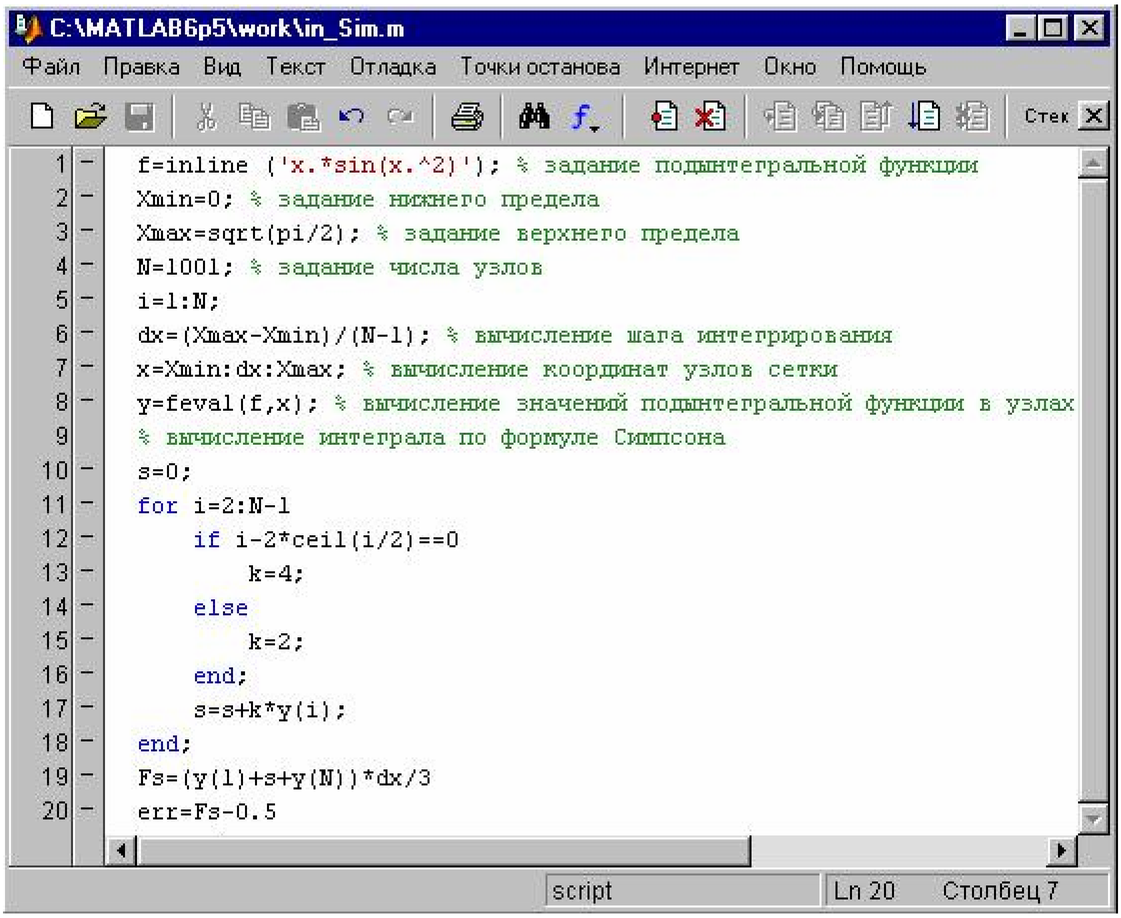
методом правых прямоугольников.



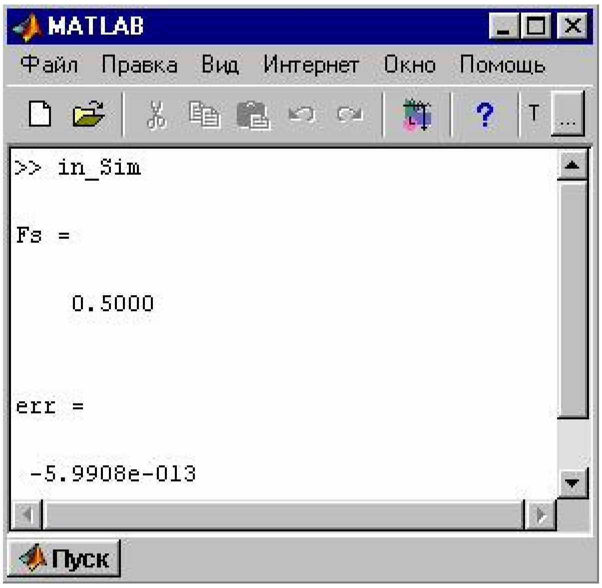
В результате работы программы получаем значение интеграла **Fr** и значение ошибки **err** между точным значением интеграла и значением, полученным методом правых прямоугольников.



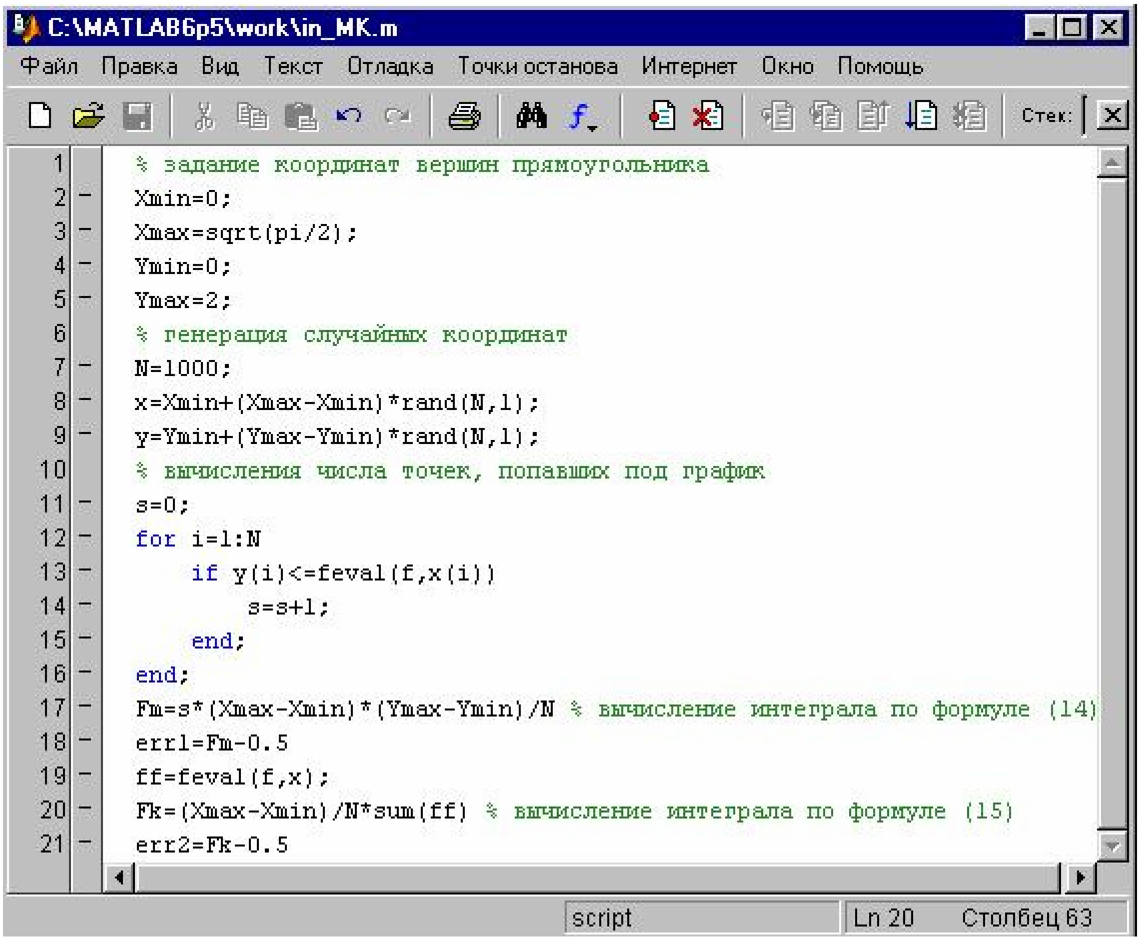
Пример 14: Вычислить значение определенного интеграла (44) методом Симпсона.



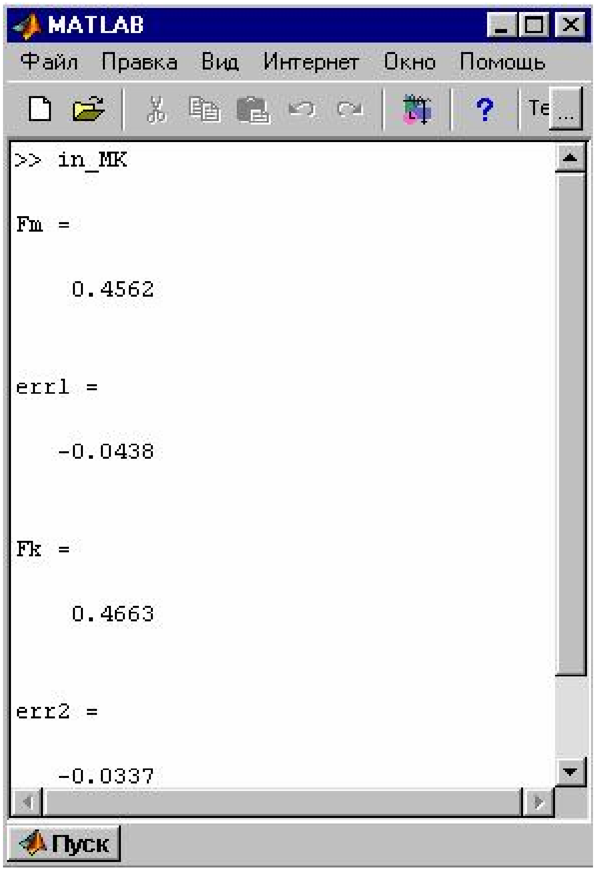
Результат работы программы:



Пример 15: Вычислить значение определенного интеграла (44) методом Монте-Карло.



В результате получаем значения интеграла, вычисленные по формулам (41), (42), и значения ошибок.



Для вычисления значений определенных интегралов в пакете Matlab применяются функции **quad ()**, **quad1 ()**, **trapz ()**.

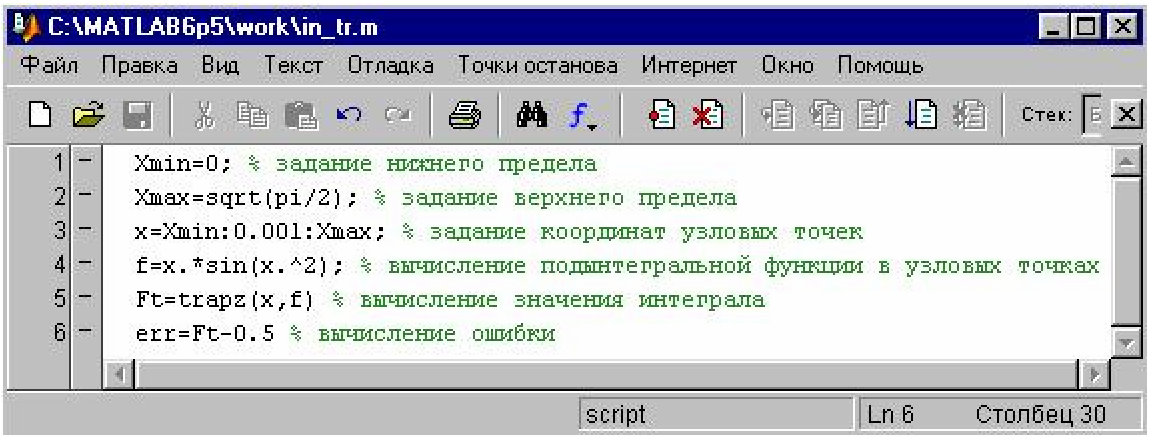
Функция **quad (*fun*, *a*, *b*)** – возвращает значение интеграла от функции *fun* на отрезке $[a, b]$, при вычислении используется метод Симпсона.

Функция **quad1 (*fun*, *a*, *b*)** – возвращает значение интеграла от функции *fun* на отрезке $[a, b]$, используя для вычисления метод Лоббато.

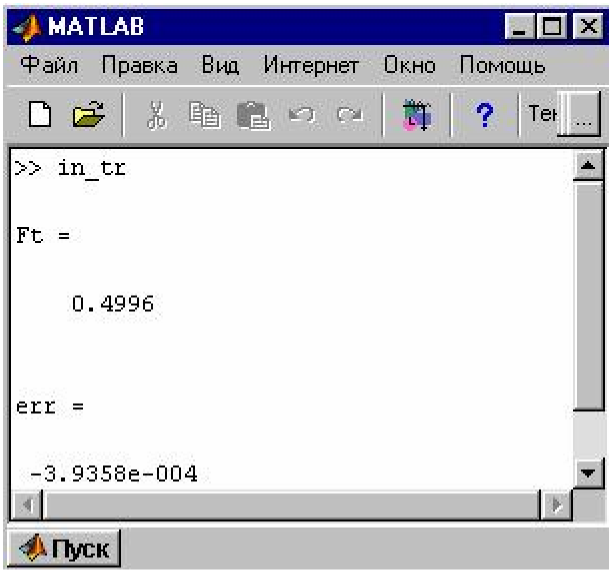
Функция **trapz (*y*)** – возвращает значение определенного интеграла в предположении, что $x = 1 : length(y)$.

Функция **trapz (*x*, *y*)** – возвращает значение определенного интеграла на отрезке $[x(1), x(n)]$.

Пример 16: Вычислить значение определенного интеграла (44) с помощью встроенной функции Matlab trapz ().



Результат вычисления интеграла представлен в окне:



Контрольные вопросы

1. Каким образом можно повысить точность численного дифференцирования?

2. В чем основные преимущества формулы трапеций по отношению к методу прямоугольников?

3. Как выбирается шаг интегрирования?

4. Чему равен порядок погрешности формулы Симпсона для двумерной подынтегральной функции?

5. Какие условия обязательно должны выполняться в методе трех восьмых и почему?

Индивидуальные задания

1. Дана таблица значений функции $y = f(x)$.

x	y	x	y	x	y
1.0	1.2661	1.6	1.7500	2.2	2.6291
1.1	1.3262	1.7	1.8640	2.3	2.8296
1.2	1.3937	1.8	1.9896	2.4	3.0493
1.3	1.4693	1.9	2.1277	2.5	3.2898
1.4	1.5534	2.0	2.2796	2.6	3.5533
1.5	1.6467	2.1	2.4463	2.7	3.8417
				2.8	4.1573

Найдите:

– значение первой производной функции в точке $x = 0.8 + i \cdot 0.2$, где i – номер вашего варианта;

– значение второй производной функции в точке $x = 2.6 - i \cdot 0.1$, где i – номер вашего варианта.

2. С помощью интерполяционных формул Ньютона найдите значение первой и второй производных функции, заданной таблично в главе 4 вашего варианта, в указанных точках

Номер варианта	x	Номер варианта	x
1	$x=1.50,\ \ x=1.51,$ $x=1.59,\ \ x=1.60.$	6	$x=0.50,\ \ x=0.51,$ $x=0.59,\ \ x=0.60.$
2	$x=0.00,\ \ x=0.05,$ $x=0.45,\ \ x=0.50.$	7	$x=1.1,\ \ x=1.2,$ $x=2.0,\ \ x=2.1.$
3	$x=1.0,\ \ x=1.1,$ $x=1.9,\ \ x=2.0.$	8	$x=0.10,\ \ x=0.35,$ $x=2.35,\ \ x=2.60.$
4	$x=0.50,\ \ x=0.51,$ $x=0.59,\ \ x=0.60.$	9	$x=1.50,\ \ x=1.51,$ $x=1.59,\ \ x=1.60.$
5	$x=0.10,\ \ x=0.35,$ $x=2.35,\ \ x=2.60.$	10	$x=0.00,\ \ x=0.05,$ $x=0.45,\ \ x=0.50.$

3. Вычислить приближенное значение интеграла по формулам трапеций и трех восьмых, разбивая отрезок интегрирования на 9 одинаковых частей. Вычислить определенный интеграл с помощью функции quad(). Сравнить полученные значения интеграла с точным.

Номер варианта	Интеграл	Номер варианта	Интеграл
1	$\int_1^e \frac{\ln^2(x)}{x}dx$	6	$\int_1^2 \frac{dx}{x^2+x}$
2	$\int_0^2 \sqrt{4-x^2}\,dx$	7	$\int_1^{e^{\pi/2}} \cos(\ln(x))$
3	$\int_0^1 \frac{x}{1+x^4}dx$	8	$\int_1^3 x^3\sqrt{x^2-1}dx$
4	$\int_0^{\frac{\pi}{6}} \frac{\sin^2(x)}{\cos(x)}dx$	9	$\int_{\ln(2)}^{2\ln(2)} \frac{dx}{e^x-1}$
5	$\int_{-1}^1 x\cdot\arctan(x)dx$	10	$\int_0^1 \frac{2\cdot\arcsin(x)}{\sqrt{1-x^2}}dx$

Библиографический список

1. Бахвалов Н.С. Численные методы / Н.С. Бахвалов, Н.П. Жидков, Г.М. Кобельков – М.: Лаборатория Базовых Знаний, 2000. – 624 с.
2. Березин И.С. Методы вычислений / И.С. Березин, Н.П. Жидков – М.: Физматлит, 1966. – Т.I. – 464 с.
3. Березин И.С. Методы вычислений / И.С. Березин, Н.П. Жидков – М.: Физматлит, 1962. – Т.II. – 640 с.
4. Вержбицкий В.М. Численные методы. Линейная алгебра и нелинейные уравнения – М.: Высшая школа, 2000. – 266 с.
5. Волков Е.А. Численные методы – М.: Наука, 1987.
6. Воробьева Г.Н. Практикум по численным методам / Г.Н. Воробьева, А.Н. Данилова – М.: Высшая школа, 1979. – 184 с.
7. Копченова Н.В. Вычислительная математика в примерах и задачах / Н.В. Копченова, И.А. Марон – М.: Наука, 1972. – 368 с.
8. Косарев В.И. 12 лекций по вычислительной математике – М.: Физматкнига, 2000. – 224 с.
9. Мэтьюз Д.Г. Численные методы. Использование MATLAB / Д.Г. Мэтьюз, К.Д. Финк – М.: Издательский дом «Вильямс», 2001. – 720 с.
10. Поршнев С.В. Вычислительная математика. Курс лекций – СПб.: БХВ-Петербург, – 2004. – 320 с.

Лариса Владимировна Чепак,
ст. преп. кафедры ИУС АмГУ, канд. техн. наук

Анна Геннадьевна Масловская,
ст. преп. кафедры МАиМ АмГУ, канд. физ.-мат. наук

Численные методы. Использование Matlab

Учебное пособие

Подписано к печати Формат 60Х84/16.
Усл. печ. л. 3,61, уч. – изд. л. 3,7. Тираж . Заказ 115.